Vic COMPUTING

Better Basic
Yet More Victuals

JUNE 1982 Vol 1 ISSUE 5

BIG EARS REVIEWED

BEGINNERS' PEEKs

MERGING PROGRAMS

the vic centre

Big savings for starters

If you're looking for a home computer, you'll already know that the VIC does it all. To help you get started, we have put together this attractive cost-saving package.

VIC-20 Colour computer

+ Cassette Deck
+ 10 Blank Cassettes
+ Introduction to Basic Part 1
+ User Manual
+ 12 months warranty
+ Fitted 13 amp plug

for only **£212**

(price plus VAT £243.80)

NEW

Convert your VIC to a 40 Column 32k machine with this exciting new add-on

32k RAM Expansion + 40 column screen format **£199.95** plus VAT

40 column screen format only **£99.95** plus VAT

Dot Matrix Printer

Tractor feed, 80 characters per line at 30 characters/seconds. **£199.96** plus VAT

Memory Expansion Board

Multi-slot unit used to accept memory and/or game and program cartridges **£85.00** plus VAT

Plug-in Memory Expansion

3k – **£26.04,** 8k – **£39.09,** 16k – **£65.17** plus VAT

Single Drive Floppy Disk Unit

£395.00 plus VAT

The standard features of the Vic are: ★ 5K RAM expandable to 32K ★ 16 screen colours ★ 8 character colours ★ 3 tone generators, each of 3 octaves – plus white noise generator ★ Screen display 22 characters×23 lines deep ★ Full PET type graphics ★ High resolution graphics capability ★ 8 programmable special functions.

Goods required _____ Price _____

_____

_____

_____

Add £4.00 post, packing and insurance for special deal and £1.00 post, packing and insurance for all other items. Add 15% to all prices for VAT

Total £ _____

Name: _____

Address: _____

_____

MAIL ORDER to: Adda Home Computers Ltd. FREEPOST, London W3 6BR or telephone your order (24 hours a day) to 01-992 9904 quoting your Visa, Access or American Express number.

*I enclose a cheque, made payable to Adda Home Computers Limited for £ .....

*Please charge my Visa/Access/American Express account. My acount number is

*Please add my name to your mailing list

*Delete as applicable

Date .....

SHOP ADDRESS: Adda Home Computers Ltd. 154 Victoria Road, Acton, London, W3. (near North Acton tube station) Tel 01-992 9904 OPEN: 10am–6pm (Tuesday-Friday), 10am–5pm (Saturday).

V

adda

TERMS AND CONDITIONS: All goods sold subject to Adda terms and conditions of sale. Full details available on request, but include: 7 day money back guarantee, Adda 12-month hardware warranty. Please allow 21 days for delivery. Allow 7 days for personal cheques to be cleared. Quoted prices are exclusive of VAT.

Contents

Under Review: the Big Ears speech recognition unit — 8

Have you ever wanted to talk to your computer? Not just to bellow curses, but actually to give it real verbal orders? Here's one solution,,reviewed by Chris Durham — a cheap and effective voice recognition unit for the Vic . . .

An introduction to PEEKs, POKEs, and the screen memory — 13

If you're only just getting into the occasionally baffling world of the Vic, our 'Beginners Please' series is for you. This contribution from Commodore's Canadian Vic supremo Paul Higginbottom will open your eyes to PEEKs and point you in the right direction on POKEs . . .

Tommy's Tips — 18

The indefatigable Tommy ploughs through another postbag of comments and queries, answering all and adding his own views along the way . . .

Merging Basic programs by Butterfield — 27

Ever wanted to put a neat routine into a program? Maestro Jim Butterfield vouchsafes some good ideas, hard facts and valuable experience on the subject of adding one Basic program to another . . .

A Matter of Routines — 48

And here are a handful of subroutines on which you might like to try those Butterfield ideas. Our regular page of useful and usable patches of code continues . . .

Be a Better Basic Buff — 16

So, you think you can write pretty nifty programs, huh? Well, Chris Preston might well open your eyes. Find out how to make your programs run faster and take up less space . . .

Todd's Lore — 23

Mike Todd bares his soul via a biography and browses around the Vic's internals to good effect . . .

Victuals — 24

Nine pages of your programs — and regular readers will doubtless be pleased to note a few innovations designed to improve their legibility . . .

DIY symbols: the Programmable Character Editor — Part Two — 30

We conclude last issue's exposition of the all-purpose editor for generation of your very own graphics and/or character sets. Part two covers the saving and subsequent use of the new symbols and shapes . . .

Dear Vic — 44

A typical collection of letters from readers — but including two items of serious note: a reprint of a Victual and a blast about stealing programs . . .

Editor
Dennis Jarrett
Workers
Stacey Tanner, Hywel David, Kevin Elyot
Subs Manager
Annabel Hunt
Ad Manager
Cathy Lane
Production
Rock Storm
Art and design
Karen the Black Douglas, Paul Hulme
Old master
Julian Allason

Telephone: 01-607 9489 (all departments)

Vic Computing is published six times a year by Paradox Group, 39-41 North Road, London N7.9DP. Annual subscription: £6 per year (UK), £IRL 8.50 (Eire); £9 (Europe), £16 (rest of world). Printing by Thamesmouth Printing Co Ltd, Southend-on-Sea, Essex; Phototypesetting by Carlinpoint Ltd, London WC2;

All editorial contributions are welcome; material is accepted on an 'all rights' basis unless otherwise agreed. Enclose SAE if you want the return of original manuscripts and photographs. Please do not send listings: we only accept programs in cassette form.

Netted

The idea of 'local area networks' is one of the hot-button buzzphrases of the computer industry right now: everyone who makes, sells or uses computers wants one. And Commodore has got one — including you, me and indeed anyone else with a Vic. Or at least we'll be able to use the KEYNET system when the Vic bit is announced in the Autumn.

What's the big deal? Well, a local network is a means whereby different computers and separate devices can all be attached to the same data-carrying link. That means, for instance, that small and cheap computers could share desirable but expensive things like disks and printers: why have one disk unit per computer? Why not share it between five or six? After all, none of them is going to have to have 100 percent access to it all the time.

It doesn't just have to be the hardware. If six people all want a particular program, you don't need to buy six copies of the software if it's there sitting on a disk that the six computers are sharing. The other plus for local networks is of course that different computer users can send messages to each other, exchange files or programs, and get at communally available information — a bulletin board, for example.

The 'local' bit, incidentally, implies that all this is happening within a fairly small area — typically one building. That's because there are technical difficulties in shifting data around at high speeds over long distances, and different types of data communications considerations apply.

Anyhow, local networks are pretty desirable. And Commodore's Keynet looks more desirable than most — though whether the "major advance" that Commodore claims is debatable. But the overall length of this network can be 1.8 kilometres, which isn't bad. In theory up to 200 individual devices (printers and disk drive systems as well as computers) can be attached to it. It is quite cheap (£200 or so per participant), it's simple to use, and it's easy to maintain.

All the participants get a special circuit board and are connected via standard four-twisted-pair telephone cable. One computer on the network has to be the 'master' (it gets a slightly different board) and in general this controls what actually happens — all the other participants are 'slaves'. Some special software runs in the master; the slaves need no special software (it's all there in a ROM chip on the Keynet board).

The slave/master approach is one of the simpler ways of implementing a local networking system — an alternative is to have no master system at all, instead putting all the control elements required in each of the interfaces on the network. There's a drawback or two to the use of a master, of course: apart from the (admittedly small) overhead of running the master programs and using up memory, there's the (more demanding) problem of what happens if for some reason the master system fails. Then your whole network can't work. Still, it's a matter of swings and roundabouts.

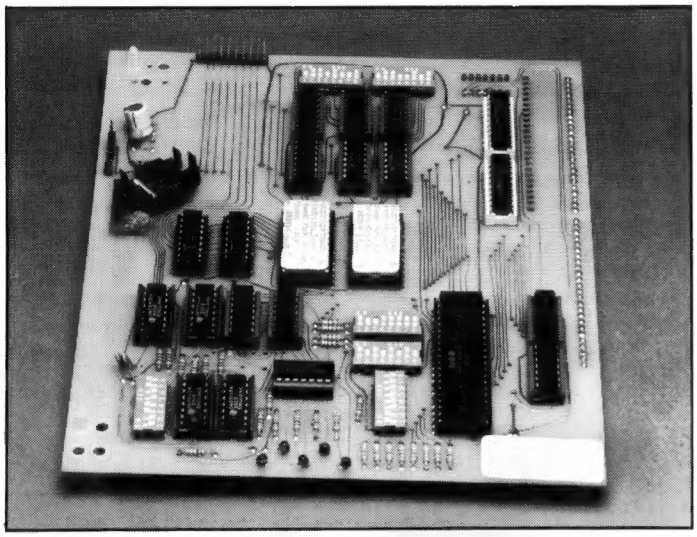
To use the network, your programs just need a couple of extra Basic commands that tell Keynet you're accessing files or using devices that aren't actually attached to your system.

Any of the standard Commodore computers can be linked on the same Keynet — each of them does need a special Keynet board, though. Already available are the versions for the old 40-column Pets (3000 to 4000 series) and the business-orientated 800s. The Vic 20 version is promised for the Autumn: we'll keep you in touch.

Mini modem

If you want to talk to another computer, you can do it by sending and receiving data over phone lines. But computers use digital signals, and the phone system (at least for the moment) doesn't. So to get information on to the line you need a gizmo to convert digital pulses into the analogue signals used by British Telecom: and at the other end there has to be a similar device to reconvert to digital.

Those black boxes are called 'modems', and over in the States Commodore recently announced the cheapest modem the computer industry has ever seen. The new Vic-1210 costs just $109.95, and that includes a program. The Vic modem is not as basic as you might expect for the price, and the VICterm program is full of good features — for instance you can ask that text typed by you is displayed in one colour while text received from someone else is shown in another.



Nifty network: one circuit card looks much like another, but at least this one bears the maker's name. Hardware for the Keynet networking system includes one of these per computer, for about £200 apiece.

This is the only way you'll get more than 20 characters per line on your Vic at present — the Beelines unit, which for its £253 also gives you another 32K of RAM.

The modem also happens to be impressivley small, plugging into the user port as it does: if the one we saw is a production version it's the same size as the Vic's ROM cartridges for games, the memory expansions, and so on.

The good news is that there's a few of the things circulating around Commodore's elegant headquarters in downtown Slough. The word is no significant technical problems have emerged on the comms side.

The bad news is that there's no easy way of connecting directly to a standard UK telephone handset, so a separate cable-connected rubber muffler may have to be added — you'd then shove your telephone handset into the rubber housing.

We understand Commodore's people are hoping for a product launch later this year at around £70.

Stacking up

Anyone who reads Vic Computing diligently from cover to cover will notice that the rearmost usually features the advertising of a company called Stack Computer Services. Stack happens to have some of the best ideas for Vic add-ons, and it's done something about them too.

The ads summarise most of them: that's a good light pen there, by the way, and we've also heard nice things about the VICKIT II programmers' aid and graphics toolbox. The latest addition is the Stack STOREBOARD memory expansion unit, which for just £49 gives you enough sockets to plug in an extra 24K of user RAM chips. There's also the sockets for a 3K wadge of high-resolution graphics memory: and one for a VICKIT or similar chip (could be a game).

The STOREBOARD comes as a neat box (with detachable cover) and plugs into the memory expansion port. It sits securely at the same level as the Vic, so you shouldn't damage the connections by having it waggle around, and no extra power is necessary for it.

Stack says the STOREBOARD was designed to enable you to get up to the Vic's maximum addressable memory capacity of 32K as and when you need (or can afford) the extra. Buy the board now, buy the additional memory when you want — Stack will sell you four chips (8K) for £39, for example.

You don't lose the Vic expander slot with the STOREBOARD, either for it is reproduced at the back of the Stack unit. Into it you can plug something like Stack's own four-slot chip board for about £25: that allows you to use more ROM chips.

Stack is on 051-933 5511. But many of the Vic dealers will be taking these new products, so try your local bloke first.

Into the 80s

If you work for Commodore you might well be fed up with all the carping about the 20-column screen.

If you don't work for Commodore you're probably doing some of the carping yourself. The solution? Well, if you're on Commodore's side of the fence you're probably advising the punters to save up for one of the new 40-column Vics (which will surely have 'official' expansion products to follow, taking the little beast up towards the Grail of an 80-characters-per-line display).

But if the Commodore is not handling your PAYE, you're getting some more immediate choices: plug in an expansion kit.

On this side of the water there's Beelines and its 40-column converter that also gives you another expansion slot for extra goodies and a helpfully large chunk of additional user memory — 32K to be precise, so you can write REALLY big programs to use your screen's new-found capacity. Still, at £235 (inc VAT and p&p and 12-month's guarantee) you might find the outlay of rather more than you paid for the Vic in the first place makes too large a dent in your bank balance.

Hang on in there, for we hear there's a nice little international industry building up for Vic add-ons now that the thing seems to be getting some promotion in Europe and the States as well as Britain — with more goodies and lower prices.

For instance, our team of diligent researchers recently uncovered one Steven E Schlanger, supremo of a small Californian outfit called Quantum Data. He's been selling Vic peripherals for a while, but his latest looks like the jackpot: it's a board that plugs into the expander slot that gives you — count 'em — 80 columns on the Vic screen.

Price? About $230, or $330 if you take the version which has another 16K RAM on the board. That translates into a not too painful number of pounds Sterling.

Sadly we don't know (a) whether it would need a UK-specific power supply; or (b) whether it will work ok with our colour TV signals (should do); or (c) whether it manages to retain the standard Vic graphics (Beelines had to opt for Prestel's rather different viewdata graphics for its own unit); or (d) whether Quantum Data has found anyone interested in flogging it over here.

But keep watching this space: teams of highly trained investigators have been commissioned to scour the Costa Mesa area to get the answers from Mr Schlanger. We can add that he's not stopping at the add-on alone, by the way. He's keen on putting on to it various software packages in PROMs, with a VisiCalc-style spreadsheet calculator and a word processor promised soon as the first of several.

So what? Well, for one thing the provision of plug-in PROMs will mean that the average Vic household can load and run lengthy and sophisticated programs as a decent 80-column word processor without having first to invest in a disk unit. You'll probably want to get a disk sooner or later, if only because loading individual files from cassette becomes a clumsy and tedious process when you have a lot of them. But with Mr S's kit you can start without the expensive peripheral.

Quantum Data will have a GT version of its board neatly packaged with power supply, cabling, the 80-column function, 16KB, and a word processing PROM for something like $700.

Add a Pet

Simple Software is one of the more amiable companies around the Pet business — well, with a name like that they'd have to be, wouldn't they? — and among other things (like excellent software) the firm puts out an occasional promotional newsletter which constitutes one of the better reads in the microcomputer business.

The latest to arrive on the editorial desk effectively promotes the use of the Vic as a terminal for the Pet. The point is that with some kind of link between the two a Pet owner can use all the Pet's facilities (notably the much more pleasant 40- or 80-column listing format, as well as all the local peripherals you might have) in order to write programs for the Vic. We would add that you could also use the Vic to collect data for Pet programs, and so on.

How to do it? Simple Software points out that you can pay £54.50 for the Vic's IEEE interface cartridge and thereby plug in a Pet directly, or use standard Pet peripherals with the Vic — but then you immediately need an expansion board (Commodore's is £130) if you also want extra memory or any other add-ons.

So as an alternative you might consider SS's £37 SIMPLY LINK package. It's the same as HES's Hescom (HES wrote the software). You get a connecting cable — five feet of it, more at extra cost — and programs for the machines at both ends of it. They give you the three basic commands: you can send an area of memory, you can fetch an area of memory, or you can send a program. Nice and simple, no?

The SIMPLY LINK code can co-exist with your own programs and operate as routines, so there's a good deal you can use it for apart from data entry and sharing peripherals. For instance, either machine could be used as a giant buffer for print spooling -- enabling you to be printing a job via one computer while you carry on using the other. Two-machine games become possible. You can store hi-res colour plots for the Vic on the Pet's disk ... And so on.

More info from Simple Software on 0273 504879.

Trading with Vic

by Ken Hall

Of all the current commercial undertakings, one in most need of computer help is the corner shop, the small supermarket, the retailer and the pub. In relation to their turnover and number of staff, they need the same amount of information as the large corporations — and the same type of information.

To succeed as a trader you need to be tough, shrewd and dedicated: but you also require up-to-date information about stock levels, cash balances, debtors versus creditors, income versus expenditure — all the things that the large companies study, helped by their accountants, clerical staff and their sophisticated computers.

So — the trader needs a computer! In the early days of a small company it is easy to keep track of everything. You can see the stock, you know all your suppliers, you soon get to know the bad payers. But, with the least bit of growth, it all gets out of hand: paperwork gets behind and usually it is the accountant who sorts out the problems.

But how much time will it take to learn to use the gear and how many hours each week to keep it all up-to-date? Money is short and it is precious, but time is in even shorter supply and so is the mental energy required to sort out and operate computerised accounts. There has to be another approach to the problem.

That mess need not happen if you keep computerised accounts. "How much?" you will ask!.

Well, you will need something of the order of 32K to keep track of a year's trading, and a disk unit to keep accurate, available information, and — of course — the programs..

Suppose that your **accountant** had the 32K computer — say a CBM 8032 with an 8050 disk unit and a printer. He could use it for example with Visicalc for his forecasting programs and Wordcraft for letters and documents: and he could have a set of programs dedicated to traders' accounts. With that set-up he could give a superb management service to the trader with monthly reports, quarterly accounts, help with stock-taking, and annual accounts produced within a few days of the end of the year.

It is acknowledged that the Vic is an excellent learning machine; but there have been very few business applications so far. However, if the trader had a Vic with an extra 16K of memory, he or she could enter each week's figures and supply these to the accountant on cassette supported by the relevant vouchers and receipts, etc. The accountant could add the data into the Pet and on to the trader's data disk ready for the production of reports and the trading account at the end of the year.

That's all right in theory — but will it work? Well a new set of programs has just been written under the name SPOTLIGHT to do just what has been described. Produced by a retired businessman, Ken Hall, and overseen by accountants SPOTLIGHT provides a tape for the Vic which will take all the details of a week's trading —. daily takings, purchases of goods for resale, business expenses, non-sale income, bank transactions, cash-in-hand and till float. The Vic screen display has been used to ask for data and show the results. Cash reconciliation is incorporated so that shortages and surpluses are shown up. According to its (entirely unbiased) author, the operation of the program has been kept as simple as possible and should present no problems for the first-time user.

If the trader could help by producing his information to the accountant at regular intervals throughout the year instead of handing over a batch of paperwork at the end of the year, the accountant could avoid the great end-of-year hassle and the computer would do the work of providing the management reports.

That's where the Vic comes in!

The accountants' package uses separate disks for each of the VAT retailers' schemes. It also has a disk with a year's trading already entered for a fictitious trader. The manual is set out in the form of a training exercise using the prepared data: the user can go through a whole year's work using every program in the suite. The program for the annual accounts, for example, can only be run after the year's data has been entered; but, using the prepared data, the results from this program can be seen as part of the running-in process.

In this way the accountant can offer monthly income/expenditure reports and quarterly management accounts including revalued stock, debtors and creditors. VAT liability can be estimated monthly and is calculated quarterly. In the case of VAT Scheme D, the year's liability is checked against the quarterly assessments.

At the end of the year the program adjusts for debtors and creditors before producing the trading account. A draft profit and loss account is printed from which it is possible to make amendments, moving sums from one part of the account to another. Those amendments are recorded by the computer and incorporated into the final profit and loss account. Figures are shown giving percentages of profit or loss and markup on takings and purchases.

Small traders are not the most profitble clients for accountants: and, at present, the accountant finds it difficult to provide the sort of management support that the trader needs. The combination of Vic and SPOTLIGHT makes this joint action possible. The trader does the donkey-work of entering the data, the accountant's computer produces the reports.

Dedicated programs of this kind will undoubtedly become widespread as the limitations of the generalised packages are realised. From a programmer's point of view, the emergence of compatible computers in different price ranges and with differing capabilities, offers great scope for development. The use of Vic as a data-receiving terminal is an obvious application..

SPOTLIGHT was written by Ken Hall, 4 St. Paul's Court, Kettering (0536-515036) and is to be distributed by Barton Services Ltd., Hall Farm, Barton Hill, Kettering (0536-81782).

SEE VIC AT THE INTERNATIONAL COMMODORE COMPUTER SHOW, CUNARD HOTEL, HAMMERSMITH, 3rd-5th JUNE.

3rd June 12 noon-6pm. 4th June 10am-6pm. 5th June 10am-5pm.

"Give me one good reason why I should choose a VIC 20 home computer."

1. VIC is outstanding value for money. No other colour home computer can give so much for under £200.

2. Total standard memory 25K made up of 20K ROM and 5K RAM.

3. Fully expandable to 27½K user RAM.

4. Microsoft Basic interpreter as standard.

5. Accessible machine language via plug-in cartridges.

6. Connects direct to monitor or standard television.

7. Full size typewriter style keyboard.

8. Full colour and sound.

9. All colours easily accessible.

10. 62 predefined graphic characters direct from the keyboard.

11. Full set of upper and lower case characters.

12. 256 displayable characters direct from the keyboard.

13. High resolution graphics capability via plug-in cartridges.

14. Programmable function keys can be used with plug-in cartridges.

15. Automatic repeat on cursor function keys.

16. User-definable input/output port.

17. Machine bus port for memory expansion and ROM software.

18. Standard interfaces for hardware peripherals.

19. VIC 20 is truly expandable into a highly sophisticated computer system with a comprehensive list of accessories (see panel below).

20. Full range of software for home, education, business and entertainment on disk, cassette and cartridge.

21. Books, manuals and learning aids from Teach Yourself Basic to the VIC programmers' reference guide (a must for advanced programmers).

22. Full support for VIC owners – their own magazine 'VIC Computing' as well as a national network of VIC user groups.

23. National dealer network providing full service and support to VIC owners.

24. Expertise and experience – Commodore are world leaders in microcomputer and silicon chip technology.

25. Commodore is the leading supplier of micro computers in the UK to business, schools, industry and the home.

26. VIC 20 is the best-selling colour home computer in the UK.

How many reasons was it you wanted?

Accessories include:

- Cassette tape unit.
- Single drive 5¼" floppy disk unit (170 K bytes capacity).
- 80-column dot matrix printer.
- 3K, 8K and 16K RAM expansion cartridges.
- Programming aid packs; machine code monitor cartridge, programmers' aid cartridge, high resolution graphics cartridge.
- Plug-in conversion box for a full 32K, 40-column x 24 lines VIC including Prestel compatability.
- RS 232C communication cartridge.
- Memory expansion board.
- IEEE/488 interface cartridge.
- Joysticks, light pens, paddles and motor controllers.

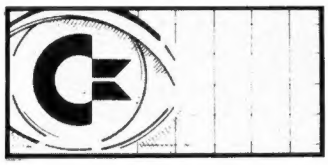
commodore

VIC 20

The best home computer in the world.

FOR MORE INFORMATION ON THE VIC 20, TELEPHONE OR WRITE TO: COMMODORE INFORMATION CENTRE, 675 AJAX AVENUE, SLOUGH, BERKSHIRE. TELEPHONE SLOUGH 79292.

Eye on Commodore



Hanoverian dynasty

Last issue we promised a report from the Hanover Fair. If you've seen the other computer comics you might have read the odd report or two from this beanfeast. But this is The Onlie True Story of How Those Stories Got Written, so hold on to your seats.

The Fair is a gigantic mind-numbing foot-numbing several-of-everything trade fair the Germans hold at the end of each April in the least interesting corner of their country — though truth to tell there was some interest generated a couple of years back when a certain editor went for a cultural visit to Hamburg and missed the last train back. He's never been the same since.

In its own way, that looks like being the result of Commodore's involvement in Hanover Fair 1982. The high-powered occasion was used by Commodore formally to announce its latest bunch of products: and the event effectively provided a statement of the company's future as well.

For one thing, on the press relations front the company went into overdrive (not to say overdose) mode. The more reputable members of the Fourth Estate were invited to take a day trip to Hanover at Commodore's expense — but most of them couldn't make it, so Commodore ended up with the collection of creatively-minded hacks who assembled in the Gatwick Hilton the night before. Dozens of freeloading has-beens and irrelevant editors had been trying all kinds of uncouth methods to bag a seat on the freebie flight, but Commodore stood firm on its guest list.

Why the Hilton at Gatwick? Because Commodore wanted to get them away to Hanover on the crack-of-dawn flight. Also because Commodore wanted to indoctrinate the press corps with a 'presentation' the night before the event.

This actually occurred over dinner and after much quaffing, so while some of us listened most attentively there was a certain amount of heckling, throwing of bread rolls, asking of long boring questions designed to show the other journalists that one knew so much more than they did, etc. This was followed by an open invitation to participate in the bar facilities thoughtfully provided by a local beauty contest, and some of the more emotional (and probably over-tired) scribblers contributed helpful cries of "Fix!" when the lovely winner was announced.

Others drained the bar of gin and retired to selected rooms with new-found friends to take part in various acts of social intercourse, typically (but not exclusively) involving the playing of cards. The Commodore contingent had of course commendably retired well before this Division Three debauchery got into full swing.

Come the early-morning calls to get us to the plane. A quick headcount revealed several greying faces, much difficulty in focussing, a widespread tendancy to place the fingers gently to the temples with a pained expression — and a crop of absentees, at least one of whom had taken the eminently sensible precaution of locking her door and taking the phone off the hook to pre-ept any attempt to stir her into motion. Commodore people were somewhat tight-lipped, especialy when it emerged that only one of the four people representatives invited from the different magazines owned by a Certain Publishing Group actually made it to the plane. Ah well, that's one of the twists of fickle fate which will inevitably bedevil any publisher that goes on record as saying that it wants to be the biggest and most successful name in computer-related magazines...

A champagne breakfast was served in flight. Many of those who had made it eagerly seized the opportunity to rstore their blood alcohol to the 50:50 level, while others accepted the offering on the grounds that it at least sounded like Alka-Seltzer.

The arrival at the Fair was distinguished by an impressive level of chaos caused by one of the journalists insisting that the party attain ingress via the Official Press Entrance. Why? Because that way you get an Official Press Pass, which gets you a free copy of the Fair showguide. Getting past the lissom Mädchen running the press gate took half an hour, with everyone having to prove their status — with varying levels of inspiration: J Allason showed a copy of the contents page of **Vic Computing,** arguing successfully that the appellation "Old Master" was a technical term associated with the production of error-free listings.

And was it worth it? One walked the length of the Fairground to the Official Press Office, had a glass of warm organge juice, collected the catalogue, and discovered it was a meaningless two-volume telephone directory of useless information built into its very own plastic suitcase because it was too heavy to carry any other way.

Then there's the Vic 20, targeted for the present as costing another £100 according to the table — but the arrival of the Sinclair Spectrum must surely persuade Commodore to drop the 20's price tag later this year.

It's also a little close to the £250 or so that Commodore dealers will be asking for the new Vic 30. The 30 is what some people (us, for instance) have been calling the Vic 40, because although it looks like a Vic 20 and uses the same Basic it displays a screen of 25 lines each with 40 characters. It gets hi-res graphics (320x200) and has all the superior sound and graphics capabilities of the Vic 10. And though it can run all the Vic 20's peripherals, it can't use plug-in cartridges from the 20 — instead it has the same slot as the Vic 10, so all of the small machine's ROM packs can be transferred easily.

There's a slight element of the best of both worlds there: the Vic 30 will get the better Basic and all the Vic 20's peripherals, while at the same time providing a kind of promise of expansion to anyone who buys the £100 Vic 10.

The Commodore 64 represents another step. it loses the friendly name 'Vic', because Commodore wants it to look a more serious proposition for business users. The 64 looks like a Vic 30, and it gets the same facilities for sound, graphics and games. But it has 64K of memory: it can run software from 40-column Pets: and there's a slot in there for a board holding a second processor.

That last attribute is the much-heralded 'all-purpose emulator'. Not many details were availbale to the distinctly tired hacks at Gatwick, but it looks as though Commodore will be announcing a plug-in Zilog Z80 on which you could conceivably run the CP/M operating system.

Then there's the new 500 range, which starts off by looking like a Commodore 64 in a restyled housing — all the same graphics and sound facilities, 64K, the second processor option, built-in interfaces for Commodore's existing Pet peripherals and for RS-232-compatible things (which means just about everything else available to computer buyers).

But the keyboard is pretty special — it has 10 programmable function keys and a separate numeric keypad — and the memory can be expanded to 256K.

Thereupon the nearest wastepaper baskets promptly filled up with unwanted copies of the burdensome document as the already exhausted pressmen straggled back down to the end of the Fairground at which they had started. There, strange to say, things did improve: we actually found the Commodore stand, had the presentations, played with the new kit, and met (some of) the important people.

The day apparently finished well with a dinner of indefinable meat-style material ("probably mastodon" as one of the assemblage put it) and a flight back (eventually — the group trooped out on to the runway first time around to find it devoid of their transport). Your own correspondent, I'm afraid, had taken the very wise precaution of catching an earlier, scheduled, flight.

The official line on what computers Commodore sells.
The new additions are in bold.

Model	Price	Availability
Vic 10	**£100**	**September**
Vic 20	£200	now
Vic 30	**£250**	**January**
Commodore 64	**£400**	**September**
4000 series	£550 to £1430	now
Commodore 510	**£695**	**September**
8000 series	£895 to £3500	now
Commodore 710	**£995**	**September**
Commodore 720	**£1595**	**September**

As press freebies go, this one was undoubtedly a costly little exercise. Was it worth it? Well, if the event has in any way adjusted Commodore's image in the eyes of the press it was probably justified.

That's because the announcements themselves do restore Commodore to the sharp end of the micro-computer business. By the end of the year it will have competitively-priced and technologically-advanced products to offer in several areas. Importantly it will for the first time be able to offer a sensible-looking upgrade path from very small computers to really quite large systems, rather than the non-compatible product lines it has at the moment. And with improved styling, good ideas like the CP/M emulator, and the promise of a hefty budget for marketing, the people who were getting a bit iffy about selling and supporting Commodore products can now breathe a little easier.

And what's all the fuss about? The new products are as previewed in the last two issues of Vic Computing, though in some cases the names and the cabinetry have been changed.

The Ultimax is now called the Vic 10, for no good reason other than marketing considerations — there's nothing '10' about it. However it does now come in a much nicer box. As we said before, it will retail for about £100. It's a games machine and a home computer ... with games cartridges, joysticks, paddles and all the expected paraphenalia of the Intellivision/Atari games machines but also a decent Basic (as a plug-in cartridge), a 40x25 display (better than the Vic 20, you'll notice), and access via programs to an excellent hi-res graphics capability and an effective music synthesiser.

And finally, let's hear it for the 700 line, a new series of machines which look like the death-knell for the existing packaging of the Pet and the 8000 line. They get an integral monitor (80x25 display), 128 or 256K memory on the models 710 and 720 respectively, the second-processor option, and two built-in floppy disk drives (wth surprisingly fast access to the information stored on them) that are styled very neatly into the cabinet.

In our view that lot all adds up to a very competitive set of products, at the low end as well as in the hotly contested market for £2,000-plus microcomputers. But where does it leave the Vic 20? Were all those people who bought one just being taken for suckers?

Well, we don't think so. But you'll have to wait for next issue's instalment to find out why.

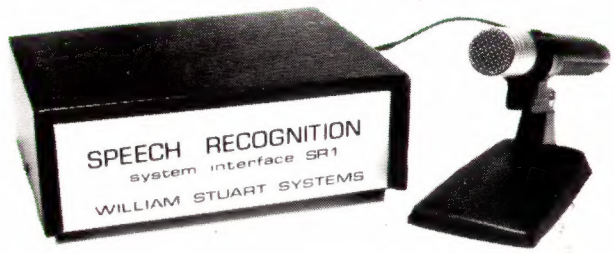
Spot the difference: this is the Commodore 64, and while it looks like the Vic 20 at this level it is in fact very different — it has 64K RAM, it gives you a 40x25 display, and it's a kind of gold colour.

A new look: the Commodore 500 starts with a spec that is not too dissimilar to the Commodore 64, and it too comes without an integral monitor. But it goes from 64 to 256KB, includes a slot for a second processor, and has a very fine keyboard.

Top of the line: this is the 720, which has two neat built-in drives and a 'full' 80x25 display — with its own tilting and swivelling screen.

Conversations with a computer: speech recognition with Big Ears

by Chris Durham



Have you ever wanted to talk to your computer? Really talk to it, that is not just yell the usual unprintable comments when your program fails with yet another syntax error, but speak actual commands to which your computer will react? If yes, this article is for you.

Speech is one of man's primary means of communication, yet we continue to communicate with our computers through strictly hand-orientated methods: keyboards, joysticks and even light pens. In the past speech has proved to be an extremely difficult method of controlling computers, but the rapid advancement of electronic technology means that virtually anyone can now have speech recognition on your home computer — especially at prices like these.

One speech recognition system that has been converted to operate with the Vic-20 is Big Ears from William Stuart Systems Ltd. This review by Chris Durham looks at what you get for your £56.35 (inc. VAT) and assesses its usefulness.

What you do actually get is a rectangular black box, a microphone with a desk stand, two leads and the instructions. The aluminium box, which measures 6 by 4.75 by 2 inches high, contains the electronics which converts speech into digital information which the computer can understand. The lid of the box is easily removed by undoing the two retaining screws: and this may be necessary since there is a variable resistor inside which adjusts the sensitivity of the unit. The unit is solid enough; and as a bonus the lid has a tasteful black leatherette finish. Nice.

There are two sockets at the back: a jack for the microphone and a five-pin DIN socket for the connection to the computer.

The microphone which is supplied is an electret condenser type. It requires a battery, which is provided: so is a windshield and an elegant desk tripod (adjustable for angle in both vertical and horizontal planes).

All this would seem to indicate that there is everything you need to make the system work. But that's not so: what's lacking is a user-port connector to connect the unit to the computer. Since you have to specify which computer you are going to use, it should not have been too difficult to add the right connector when the unit is despatched. As it is, if you don't happen tohave a spare user-port connector lying about, you have to order one specially and solder the cable to it before the system can be used at all. Presumably the cost of connectors varies considerably between different computers, making it difficult to give a standard price; but at last some mention ought to be made in the adverts to indicate that a connector will be required.

The software

The main documentation is quite long: five pages of instructions, including the 'theory of operation', plus three pages of Basic program listing. The program is well computed and can be understood by anyone with a reasonable knowledge of Basic.

There are actually two listings, one for a program that is common to all computers and a separate machine-dependent listing which also includes any changes to the main program.

The latter contains the machine-code routine which actually reads the data from the unit. To anyone who gives up at the mere mention of machine code let me hasten to add that all the hard work is done for you. The machine-code is all in the form of DATA statements and is typed in exactly the same way as the rest of the program.

Because the amount of memory is so easily changed on the Vic, detailed instructions give all the changes necessary for any combination and size of memory. The only restriction is that the program cannot be used on the unexpanded Vic, since it requires approximately 5K bytes to itself — that includes storage for up to six workds or phrases. Also included in the documentation is a list of the machine code mnemonics and addresses, which is well commented: this alows a newcomer to 6502 machine code to understand much of how it works.

Testing the system

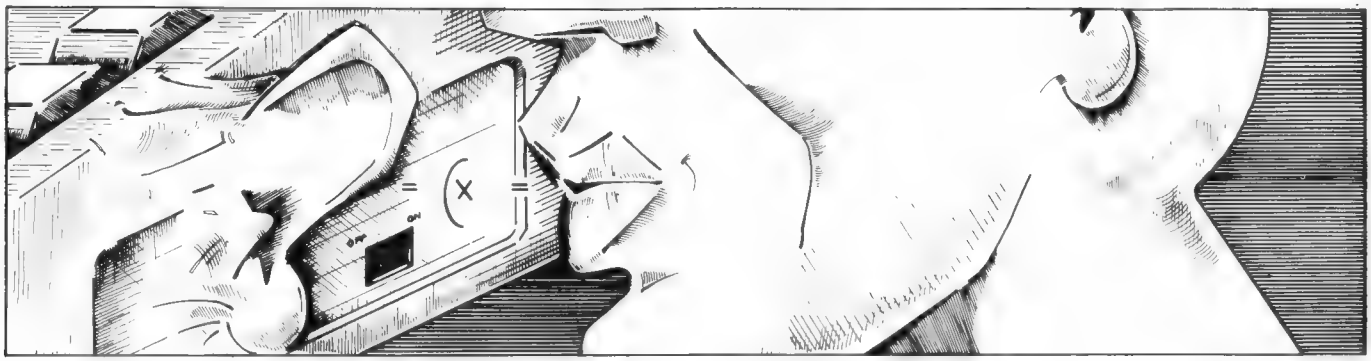
After typing in the program and checking it carefully it should be stored on tape before running it. Any program which includes machine-code routines can cause the Vic to go into a coma if there is any error in typing it in. This doesn't harm the Vic in any way, but switching off and then on again means that the program will be lost and has to be re-entered — a real drag if the program isn't saved initially.

Incidentally, there is a checksum run automatically on the machine code. The word ERROR appears when the program is run if a mistake has been made in typing the DATA statements.

Once the program is running correctly you are given the option to 'Learn or Test'. Basically, you have to teach the computer the words or phrases you want it to recognise later on. This consists of typing in the phrase and then saying it up to eight times (the number of repeats can be varied) from which the computer will store an averaged voice-print.

The test facility enables you to check that the computer can successfully recognise the word you have just taught it. You say the word(s) when prompted, the computer thinks about it for a few seconds, and it gives a score to each phrase learnt so far; this it displays. It then selects whichever phrase has the highest score. Hopefully it is the one you just said. I say hopefully, because some phrases, which sound different to you or I, can produce very similar voice-prints: the computer will often mismatch such phrases, giving rise to some rather amusing answers.

There is a limit to the number of voice-prints which the program can effectively search and match. This maximum is about twelve; but the instructions do explain how to expand this by swopping blocks of voice-prints on a cascading-menu basis and thus making the range of words limited only by the amount of memory available. Each voice-print occupies about 130 bytes, using floating point arrays, so you can work out the number of words you can store in your own machine. Attempting recognition from more than twelve phrases at a time means a very long wait while the voice-prints are checked plus a greter possibility of an incorrect recognition.



Applications

As received, the program is of little practical use since it has no facility to store the voice-prints on tape: they are lost when you end the program. So one of the first things you have to do is add the code necessary to create a data file and store the voice-prints and associated text. Once you have taught the computer the total number of phrases required you are ready to make use of the system in your own programs.

By extracting the relevant recognition procedures and the machine code routine you can add these to your programs quite easily. Instead of looking for a keyboard input, the computer can jump to the machine-code routine and wait for a speech command instead.

Since the recognition procedure returns the number of the voice-print (assigning them in the order in which the word was learnt) you can use that value as a subroutine selector: for example, ON VP GOSUB 100,200,300. Similarly it could be an array subscript or even a function variable.

Your own program only has to load the voice-prints from the data file and this saves you having to go though the process of teaching the computer each time you run your program.

The program will correctly recognise only phrases spoken by the person who recorded them. But different people who want to use the same program can each create a data file with their own voice-prints. They then load the correct tape when they want to run the program: and the computer can thereby be made to recognise several people's voices.

Specific applications will depend very much on the individual, but any program which has a menu selection or a choice of answer can be easily adapted to accept voice control. It is extremely satisfying to say "repeat" to your computer and watch it produce a screenful of data.

Any process where you are doing two things at once, such as reading a test-meter and then typing the results, can benefit from having a non-contact input method. Adventure-type games, for instance where you select a tunnel out of a cavern, make a choice of weapons or decide whether to attack or retire, are ideal for voice control. Programs like Space Invaders or Rally, which require extremely fast reaction, are not so suitable; as written the recognition procudures just aren't quick enough.

Drawbacks

There are two major drawbacks to using most speech recongition systems and Big Ears is no exception. The first is that when the microphone is live and the computer is waiting for an input, **any** noise will be accepted. So if there is any major background noise, or if you drop your pen, if someone sneezes, the program will try and match that sound to the stored voice-prints — it will obviously fail, but will come up with the best match it can. Coughing during an Adventure game could well result in your attacking a fire-breathing dragon armed with a toothbrush (or something equally inappropriate — suggestions on a postcard, please).

One way round this is to record these sounds and make the program ignore them and go back to the listening mode again. This of course reduces the number of wanted phrases you can store; there are more complicated methods of recognising spurious sounds, though; if you like experimenting.

The second significant drawback is the time taken to recognise words. The program will not recognise correctly any sound that lasts longer than one second. A short phrase like "Hello Vic" can be stored as one voice-print, as can any single words. A phrase like "Pick up the lamp" might just make it, but "pick up the lamp and rub it" would definitely not.

The program will match a single voice-print from a choice of five in about two or three seconds. Matching from a choice of twelve can double that time. And if you think six seconds or so is quite fast, say a word and then count slowly from one to six. You will quickly realise why such a response time would be unacceptable for something like Space Invaders.

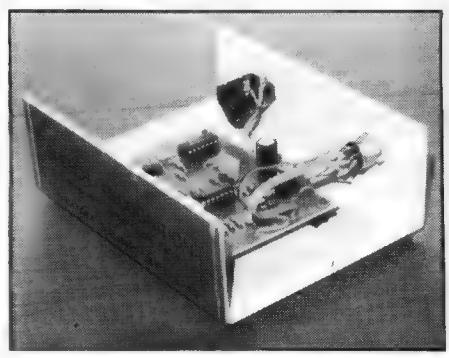
The delay is because the recognition procedures are written in Basic, with a large number of calculations and comparisons. In machine-code the problem would largely disappear, and most responses could come down to within one second. What's more, by extending the machine-code input routine slightly, multiple phrases could be recognised with each 'seconds-worth' of speech being stored separately. This is an area where anyone interested in voice-recognition could really go to town.

Conclusions

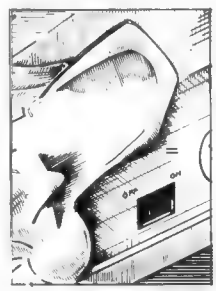
Big Ears is a cheap introduction to the fascinating world of speech recognition. It opens the way for Vic owners to experiment with a new way of communicating with their computer, a method which can be readily adapted to existing programs. There is great potential for anyone interested in machine-code to really improve the existing software and overcome the two major drawbacks. The system will also appeal to the beginner who would like to learn more about a subject which has great potential as a control method of the future.

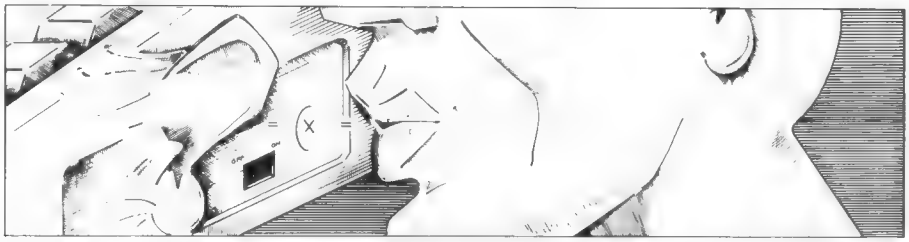
Speech recognition, with its current drawbacks, is not going to revolutionise home computing overnight. But its novelty, appeal and usefulness in specific applications will ensure that there is scope for Big Ears even at the low end of the home-computer market.

Inside the box: a lot of room, not much spaghetti



3ig Ears is available from William Stuart Systems, Dower House, Billericay Road, Herongate, Brentwood, Essex, CM13 3SD.





The business end, complete with neat mike and Big Ears' more official title

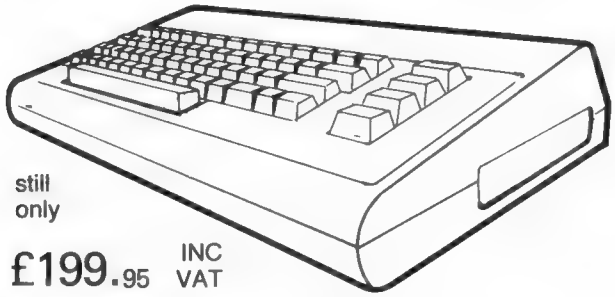
Technical summary

Physical:	Cabinet 6x5x2ins Vinyl leatherette-clad steel surround with brushed aluminium facia, non-slip feet. Weight 0.35 kg. Microphone provided.
Connectors:	5-pin DIN socket (to computer) 0.25 ins mono jack (for microphone). Din cable supplied for make-up to appropriate computer connector.
Interface:	Two bits of any parallel input port (PIO, VIA or logic storbe; handshake not used).
Memory:	Minimum 5K bytes RAM for Basic program, acquisition code and data storage for six voice-prints. Additional voiceprints occupy approx 130 bytes each if floating point arrays are used, 40 bytes each if 8-bit integer arrays used.
Software:	Supplied as machine code for real time acquisition and Basic program for analysis. Machine code is called by Basic program as a USR Subroutine.
Price:	£49 plus VAT (includes postage and packing).
Supplier:	William Stuart Systems Ltd. Dower House, Herongate, Brentford, Essex.

Sumlock
MANCHESTER.

SUMLOCK MICRO VOUCHER SUPPLIED WITH EACH VIC 20 WORTH £25 IMMEDIATELY REDEEMABLE AGAINST ANY GOODS

Microcomputer World



still only

£199.95 INC VAT

andiogenic stockist

		(inc. VAT)
VP 030	Space division	6.99
VP 032	Invader fall	6.99
VP 034	Star wars	6.99
VP 045	Math hurdler, Monster maze	7.99
VP 046	Seawolf, Bounce out, VIC Trap	8.99
VP 047	Cock maker, code breaker	6.99
VP 010	Amok	6.99
VP 011	Simple Simon	6.99
VP 012	VIC ale	8.99
VP 016	A maze-ing	6.99
VP 023	Masterwits	6.99
VP 024	Kiddie Checkers	6.99
VP 025	Wall Street	6.99
VP 026	Alien blitz	7.99
VP 029	Sky Math	6.99
VP 044	Hangman, Hangmath	7.99
VP 014	Spiders From Mars, cartridge game	24.95
VIC 1530	cassette deck	44.95
VIC 1515	printer	230.00
VIC 1540	single floppy drive	396.00
VIC 1210	3K ram cartridge	29.95
VIC 1110	8K ram cartridge	44.95
VIC 1111	16K ram cartridge	74.95
VIC 1906	Alien ROM game	19.95
VIC 1907	Super Lander ROM game	19.95
VIC 1904	Super Slot ROM game	19.95
VIC 1212	Programmers Aid cartridge	34.95
VIC 1211A	Super Expander High resolution cartridge	34.95
VIC 1213	Machine code monitor	34.95
VIC 1311	Joystick	7.50
VIC 1312	Paddles (pair)	13.50
VIC 1901	Avenger cartridge	19.95
VIC 1909	Road Race cartridge	19.95
VIC 2801	Learn computer programming with the VIC	1.95
VIC 2803	VIC 20 programmers reference guide	14.95
VIC 2501	Intro to basic part one	14.95
VIC 20	Games pack 1 (4 games) Putting, Death mire, Death race 2001, hailstones	6.99
VIC 20	Games pack 2 (2 games) Munchies, frog	6.99
VIC 20	Games pack 3 (4 games) 3d Alien maze, Death race, Monsoon, 3D doodles	6.99
VIC 20	Games pack 4 (1 game) millipede	6.99
VIC 20	Games pack 5 (1 game) gunfight	6.99
VIC 20	Games pack 6 (1 game) Man on the run	7.50
VIC 20	Games pack 7 (1 game) Space bomber	7.50
VIC 20	Games pack 8 Chopper	7.50
VIC 20	Mind games (4 games)	5.99
	Arfon expansion unit (seven slot unit with power supply and lid)	107.00
	Set dust covers for VIC 20 and cassette deck	3.75
C12	blank cassettes	0.63

TEN FREE PROGRAMS WITH EACH VIC 20 AND VIC 20 CASSETTE DECK PURCHASED TOGETHER

All items are ex stock. Telephone/mail order despatched within 24 hours. Carriage free U.K. mainland on orders over £10: under £10 add 50p p&p

COMPLETE LIST OF HARDWARE AND SOFTWARE AVAILABLE BY REQUEST

keep up to date - join our free mailing list

061 834 4233

Sumlock Manchester
Dept MC1
Royal London House
198 Deansgate
Manchester
M3 3NE

CREAM COMPUTER

380 STATION ROAD
HARROW, MIDDX.
HA1 2DE

Tel. 01-863-0833

'DISCOUNT PRICES ON QUALITY PRODUCTS'
THE VIC-20 CENTRE

HARDWARE

VIC-20 Computer (Call now for new low low price)	
C2N Cassette Unit	39.99
1540 Floppy Unit	375.00
1515 Printer	214.95
Memory Expansion	110.75
3K Ram Expansion	27.50
8K Ram Expansion	39.98
16K Ram Expansion	68.99
Super Expander Cart	32.50
Progs Aid Cartridge	34.50
Vicmon Cartridge	30.00
Joystick	6.99
3K Store Board	51.00
8K Ram Chips	39.00
Vickit I	25.00
Vickit II	29.00
VIC-20 Dust Cover	2.00
C2N Dust Cover	1.70
Light Pen	25.00
Rom Switch Board	42.00
Expansion Board	94.99

* NEW * NEW * NEW * NEW *
PRINTERS FOR YOUR VIC-20

SEIKOSHA PRINTER - The Commodore Printer plugs directly into the VIC-20 uses 8" continuous paper up to 80 chars per line with a speed of 30 c.p.s
PRICE 214.95

EPSON MX80 F/T PRINTER - Uses standard A4 paper or 9.5" paper. 80 characters per line with a speed of 80 c.p.s.
PRICE 458.99
(INCLUDES VIC-20 INTERFACE)

* * * * NEW * * * *

RABBIT WRITER (Wordprocessor)
Requires 8K or 16K Ram, includes * mailmerge * left and right hand justification * tape or disk format * prints up to 80 columns * uses 80 columns * uses all function keys for editing etc.
INTRO PRICE TAPE 9.99
DISK 12.50

RABBIT BASE requires 16K RAM
Personal information master, ideal for mailing, personal records, stock control and filing.
INTRO PRICE TAPE 11.99
DISK 13.99

* SPECIAL OFFER *
* SPECIAL OFFER *
VIC-20 PRINTER + 16K RAM PACK + RABBIT BASE + RABBIT WRITER
ALL THIS FOR ONLY 290.00

DON'T FORGET TO SEND OR PHONE FOR YOUR FULL FREE CATALOGUE

RABBIT SOFTWARE

Space Storm	6.99
Ski Run	4.99
Dune Buggy	4.99
Super Worm	4.99
Jungle	4.99
Cosmic Battle	4.99
Frogger	7.99
Rab Functions	4.99
Star Wars	6.99
Amok	6.99
Blitz	4.99
Invader Fall	6.99
Masterwits	6.99
Kiddie Checkers	6.99
Simple Simon	6.99
Alien Blitz	7.99
Functions	4.99
Code Breaker	4.99
Night Flight(3K)	4.99
The Alien (3K)	7.99
Amazing (3K)	6.99
Skymath (3K)	6.99
Spc Division(3K)	6.99
Charset 20 (3K)	4.99
Rabbitwriter(16K)	9.99
Rabbit Base(16K)	11.99
Rabbit Chase	4.99

ALL RUN ON BASIC MACHINE EXCEPT WHEN STATED

RABBIT CHASE
Rabbit version of the famous arcade game 'Gobbler'. Chase the Rabbit through the carrot fields, eat the lettuces and chase the farmers. Use Joystick or keyboard.
ONLY 4.99

Please put me on your mailing list for lots of new information on the VIC-20

Name _____

Address _____

_____

_____

All prices are inclusive of VAT and packing.

and please send me

QUANTITY	DESCRIPTION	PRICE

I enclose my *cheque/charge my Access/Trustcard/
Barclaycard/Visa No. .....
TOTAL
Signature _____

CARD HOLDERS PHONE YOUR ORDER
01-863 0833
*Delete as applicable

An introduction to PEEKs, POKEs and the screen memory

by Paul Higginbottom

At the heart of every computer there is a central processor, and your Vic is no exception. Every microprocessor understands its own type of instructions; and this unique code is what is called machine language. To put it more precisely, machine language is the only programming language that your micro understands. It is the language of the machine.

But if machine language is the only language that your machine comprehends, how come you use the Basic programming language? That isn't machine language: so how does the microprocessor understand the Basic statements that you type in?

Paul Higginbottom, who is Commodore's resident Vic expert in Canada, introduces the way the Vic's screen memory works and leads you gently into the world of PEEKs — what they are, why your Vic has them, when you can use them.

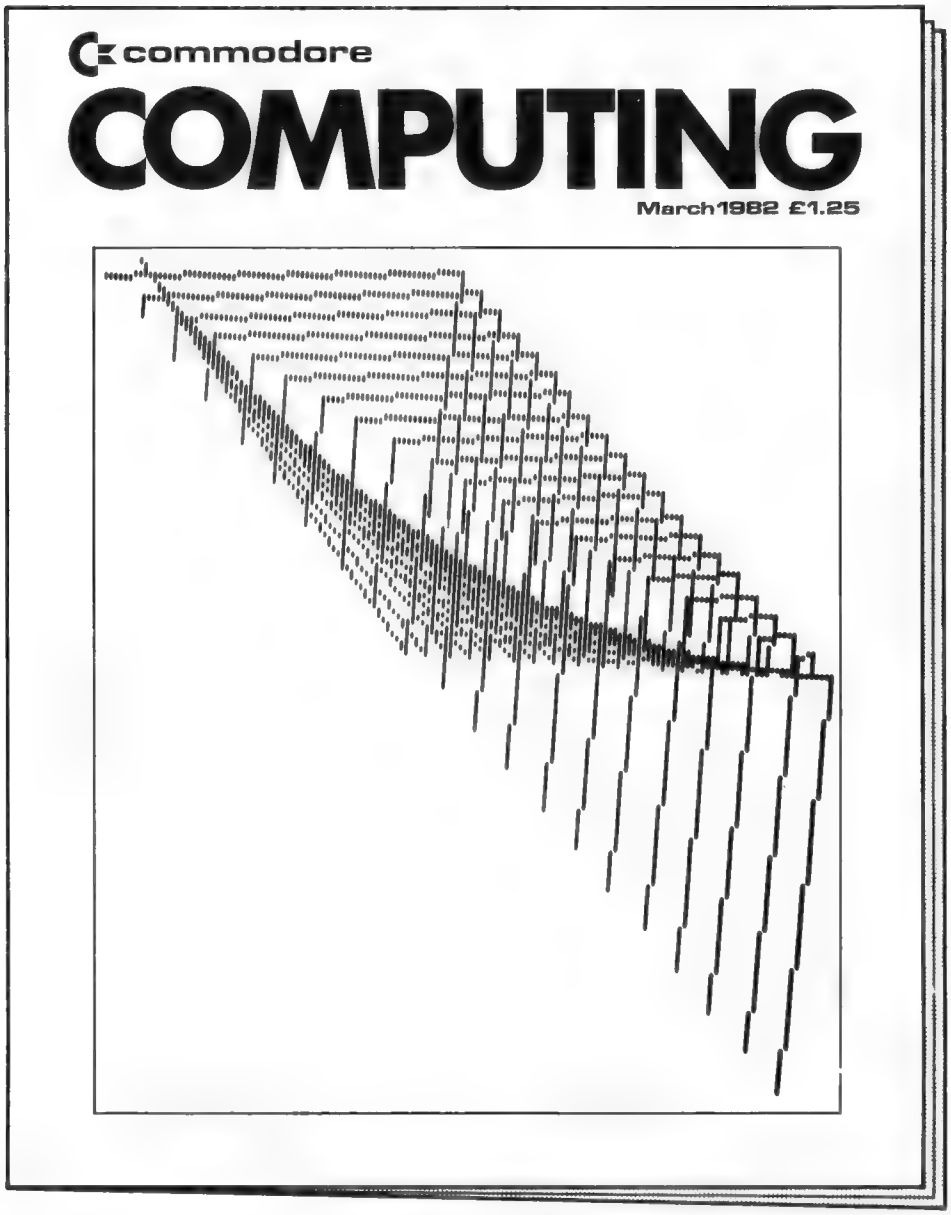
So what exactly is the relationship between Basic and the machine language?

Well, when you turn on your Vic it begins to run a huge machine-language program that is already permanently resident inside it — it is not lost when you turn off the power, and it cannot be changed. This program checks how much memory is in your Vic, clears the screen, puts the message "**** CBM Basic V2 ****" on the screen, tells you how much memory you have, and finally says 'READY'. It then blinks its cursor to let you know that all systems are go. (It also does a number of other setup jobs, but these needn't be covered here.)

When you type in Basic commands, they are in fact being understood the same machine-language program. This is why we call the Basic in the Vic-20 'interpreted' Basic: the machine language is 'interpreting' the Basic commands, one at a time. Machine language is difficult for the first-time computer user to grasp, and the interpreter allows you to write programs in a language — Basic — which is significantly nearer to English.

This is all very well until you discover such commands in Basic as POKE, PEEK, SYS, USR, and WAIT — because these commands are the bridge between the unconcerned life of line numbers and English-like programming commands on the one hand, and the nitty-gritty of the computer's on the other. Since they allow you to step from the outer world of Basic into the underworld of the internal workings of your machine, these commands are potentially lethal in computing terms: once you have the power to interfere with the way that your machine works, you also have the power to make you machine not work.

The nice thing about a 'high-level language' like this is that you don't have to worry about any concepts of memory organisation or indeed anything machine-orientated. Machine languages are specific to a particular computer, but the Basic language is much the same on most computers with comparitively minor extensions or restrictions being the differences.



An exciting new Commodore peripheral

"Commodore Computing" is the only publication that's specially for those with PET or VIC computers.

It not only keeps up with the latest developments in technology but shows how they apply to your particular system.

There will be articles by the world's leading computer authorities. Details of new products.

As well as all kinds of different ways you can get the most from your Commodore computer.

An annual subscription for 10 copies of "Commodore Computing" only costs £12.50. Just fill in the coupon for more details or call into your local Commodore Dealer. Either way—it's the Commodore peripheral you shouldn't be without.

Send to: Commodore Computing, Subscription Manager
Magsub, OAK Field House, Perrymount Road,
Haywards Heath, Sussex RH16 3DH

Please send me full subscription details of "Commodore Computing".

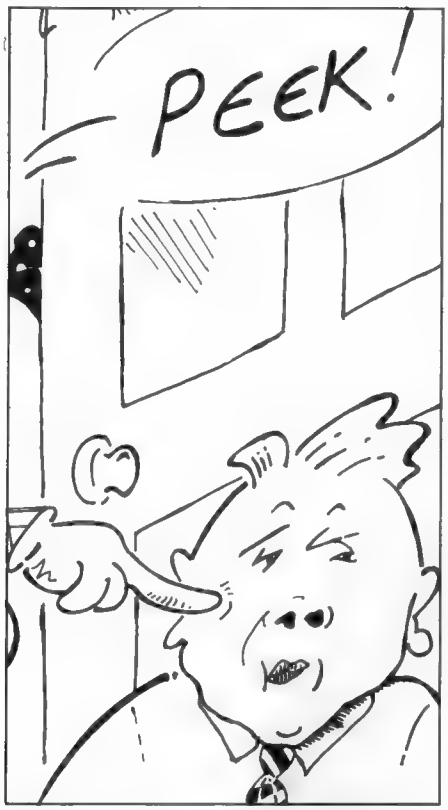
Name _____

Address _____

_____

_____

Tel: _____ 28/EC/1



One thing I must make clear is that you cannot physically hurt your Vic by using them. And to make your machine 'not' work is simply a temporary thing; it can be restored by turning the power off and then on again. The only thing you may seriously damage is your peace of mind. If you have spent two hours programming, and then 'crash' your program through experimentation, you may also pull large lumps of hair out of your head.

But although those commands may be potentially dangerous, they can be extremely useful. For instance, the Vic has a screen which can be filled with characters. These all have their own individual code to distinguish them from each other.

You may know the CHR$ and ASF functions in Basic; they allow you to see what character a given code produces — for example, PRINT CHR$ (65) — and to see what code a given character has — as in PRINT ASC("A").

Now, the picture on your television is a reflection two areas of memory in Vic — one which says which character is in each position on the screen, another that specifies what colour each character is.

The POKE command allows a memory location to be set to a particular value, and the English way of stating the POKE command would be "poke into a particular memory location number the specified value". The PEEK command will give the value contained in a memory location. Since the screen is simply a reflection of a piece of memory, we can modify the screen and also look to see what is on it by using the PEEK and POKE commands.

On an unexpanded Vic-20, or a Vic with 3K of extra memory, the screen memory begins at memory location 7,680. The character codes that can be found from the ASC function are not the same as those used in screen memory; so you will have to find the code for the particular character you are looking for from the manual supplied with your Vic.

Turn on your Vic and type CLR followed by POKE 7680,1 and RETURN. You should see that the blue 'P' of the POKE command you typed has turned into a blue 'A'. This is because the character 'A' has a screen memory code of 1; and you have put a 1 into the first memory location of the screen, the top left corner. As the memory location number (the 'address') increases, the position goes across the top of the screen and then 'wraps around' at the end on to the start of the second line of the screen.

Once that is appreciated, we can go on to understand the colour control of each character. On a basic Vic or with the 3K memory add-on, the table in memory that decides on the colour of each character on the screen starts at address 38400.

So if you now type POKE 38400,7 you should see the blue 'A' turn yellow. This is because 7 is the colour code for yellow, and we have set the colour of the first location on the screen to 7.

As with the screen memory, the colour control table goes in sequence in memory — that is, across the screen and wrapping around to the next line.

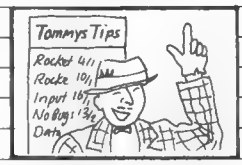
Now that we can display characters on the screen, and set their colour by POKEing, we should be ready to delve further. If we want to implement a row and column system for changing colours and characters, rather than having to know the addresses of all the locations on the screen, we could set up a neat pattern of POKEs. There are 22 columns on the screen. If the top left location is identified as column no. 0 in row no. 0, we can compute the actual memory address of any row and column position by the formula:

P=7680+Y * 22+X

– where P is position, X is the column, and Y the row.

Take a look at the programmed example to see how it works — the REM statement allows you to put remarks into a program listing thus making it easier to understand. Meanwhile try some experiments of your own: it is the easiest way to find out about your Vic. Remember, you can't hurt it — you can damage only your own pride and tempor.

```
100 REM SHOW CHARACTERS ON THE SCREEN
110 REM IN DIFFERENT COLOURS
120 PRINT "□": REM CLEAR THE SCREEN
130 COLOUR=0:CHR=0
140 FOR Y=0TO22
141 REM: ROWS 0 TO 22 TO FILL
150 FOR X=0TO21
151 REM: COLUMNS 0 TO 21 TO FILL
160 P=7680+Y*22+X
161 REM: CALCULATE POSITION
170 POKE P, CHAR
171 REM: PUT CHARACTER AT POSITION P
180 C=38400+Y*22+X
181 REM: CALCULATE COLOUR TABLE ADDRESS
190 POKE C,COLOUR
191 REM: SET THE COLOUR RUN
200 COLOUR=COLOUR+1
201 REM: COLOUR CODE IS NOW ONE MORE
210 IF COLOUR>7 THEN COLOUR=0
211 REM: KEEP 'COLOUR' BETWEEN 0 AND 7
220 CHAR=CHAR+1
221 REM: CHARACTER CODE IS NOW ONE MORE
230 IF CHAR>255 THEN CHAR=0
231 REM: KEEP 'CHAR' BETWEEN 0 AND 255
240 NEXTX: REM GO ON TO THE NEXT COLUMN
250 NEXT Y: REM GO ON TO THE NEXT ROW
260 END
```



Tommy on electronics — plus more answers

Tommy likes problems about Vic-matters. His idea of fun is to watch the postman struggling up the steps of Tommy Towers, labouring under the weight of a postbag bulging with readers' queries. It's not that he has anything against postpeople, he just loves getting stuck into your technical questions. So keep 'em coming!

One thing which has become clear already is that there are a large number of people out there who are very keen to connect their Vics to amplifiers, sound-to-light systems, synthesisers and all kinds of other electronic gadgets. The trouble is they are not sure how to do it and (quite naturally) they write to me for advice.

The trouble is that building electronic devices is not like programming. First, you can very easily kill yourself; and second, you can also very easily blow up equipment costing many hundreds of pounds.

You cannot learn how to design hardware by reading an article in **Vic Computing** unfortunately, much as we would like to do all the Universities and Polytechnics out of a job. I could spout pages of instructions about DACs and ADCs, analogue switches, operational amplifiers and so on ad infinitum but it would not do any good; like the Rolls Royce salesman who said "If you need to ask how much it is, you can't afford it". I am going to say "If you need to ask how to do it — you can't". I do hate to dampen enthusiasm, but I think safety comes first. If you are really keen, enrol in a evening course at your local college and learn how to do it properly.

There are several readers who have written to me for details of various add-on items for the Vic saying "I could build it from the circuit diagram". Yes, I am sure you could; but could you test it? To troubleshoot a computer circuit requires some fairly expensive test equipment; an ordinary enthusiast is not likely to have a storage oscilloscope or logic analyser for example.

And now after all that gloom and despondency, on to some questions I can help you with:

Dear Tommy, Is there any way to stop the cursor from flashing, keeping it on all the time?

I am afraid that is not possible to change the cursor display on the Vic. The flashing cursor, like the rest of the screen display, is produced by the video circuitry under the control of a chip called a Video Interface Chip. The particular chip used in the Vic is a 6560, which only has one type of cursor. Some chips, like the Motorola 6845, offer a range of cursors, such as a box or underline, which may be constant or flashing.

However, you can program a similar effect when your program needs to input data. Instead of using the Basic INPUT statement, you will have to write a subroutine based on the GET statement. By keeping a "cursor position" in a variable, and printing that character in reverse video, you will get what looks like a constant cursor. Whenever the 'cursor left' or 'cursor right' keys are pressed, you will have to alter the character under the cursor to normal video and change the next one to reverse. If you like, you can build all sorts of bells and whistles into your routine, disallowing various keys and changing the effect of others. It all depends on how much time you want to spend on it of course, but you can end up with an input routine which is miles better than the INPUT statement.

Dear Tommy, I have a question which will be of interest to your readers. How can I erase tapes? I wish to sort my programs on to different tapes because some use more RAM than others. On a tape recorder I just press "play" and "record" and it will erase. Then I can just record over that part again without problems. Is this OK, or is there a better way? Finally, could you give Commodore a kick up the backside, as I have been waiting for months to get the Programmers Reference Guide **which is still not available.**

Yes, it's quite all right to re-record your program tapes, so long as you do not record over something you want! You must be careful, though, if you are saving several programs on one tape. If you re-save the first program on the tape, it may have grown so that it will overwrite the second program! It is best to save each program on a different tape if you can afford it.

Another tip (which is even more expensive on tapes) is to have **two** tapes for each program and save on to each tape alternately. This means that if you find you have lost an important line, or made some changes which you now find are wrong, you have a backup copy which you can use to recover from.

As far as the **Programmer's Reference Guide** is concerned, all I can say is that you are not the only person yet to receive their copy! Can I recommend **The VIC Revealed** by Nick Hampshire? It contains oodles of useful tips and information for us enthusiasts.

Dear Tommy, I have two queries which may affect other owners. Should the Vic give timebase flyback lines at the top? These are faint but annoy me.

If the flyback is visible on your TV or monitor, it is the monitor which is producing it. The output from the Vic is a brightness signal (colour-coded) which varies as the spot moves across the screen to produce the image.

However, if the output from the Vic is of low amplitude, you will have to turn up the contrast control on the set to compensate; so the Vic could be indirectly at fault. If you are affluent (or is it effluent, I can never remember. Sorry if I've got it mixed up!) enough to have two televisions, try the Vic with the other one. If not then it is best to whip the Vic back to your dealer. He will (probably) be glad to try it out on one of his monitors (and no doubt try to sell you some more software while you are there).

Dear Tommy, Over the last few days I have been ripping my hair out trying to display a running clock at the top of the screen while a game is running. I have tried POKEing and PEEKing and even simple PRINT statements without success.

Tut, tut, I am ashamed of you. Pulling your hair out indeed! There are some of us who are desperately trying to stop our hair falling out — please consider our feelings.

Anyway, to your problem. If all you want to do is to display a clock while waiting for the user to press a key, say with a line like:

```
100 GET A$ : IF A$ = "" THEN 100
```

then you can get away with the simple solution:

```
100 GET A$ : IF A$ = ""THEN PRINT CHR$(19);TI$:GOTO100
```

If on the other hand, you need the clock ticking away all the time, even when your program is chuntering away doing calculations, we need to be a little bit more subtle: and of course that means using machine code. Every 1/60th of a second, the VIC stops what it is doing (e.g. running your Basic program) and performs some essential "background" operations, such as updating the jiffy clock and scanning the keyboard. What we need to do is to insert a little extra bit of code into this "interrupt routine" to dislay a clock on the screen. This is for a standard, unexpanded VIC by the way.

```
100 POKE 55,29 : CLR
110 DATA 120,169,28,141,20,3,
    169,29,141,21,3
120 DATA 169,0,141,118,29,169,
    48,160,5,153
130 DATA 112,29,136,16,250,88,
    96
140 DATA 238,118,29,173,118,29,
    201,60,208
150 DATA 34,169,0,141,118,29,
    160,5,56,185
160 DATA 112,29,105,0,201,58,
    208,3,169,48
170 DATA 56,153,112,29,153,0,
    30,169
180 DATA 150,153,0,150,136,16,
    230,76,191,234
190 DATA 120,169,191,141,20,3,
    169,234,141,21,3,88,96
200 FOR I = 7424 TO 7511 : READ
    J : POKE I,J : NEXT
```

The Clock is turned on by a SYS 7424 and off again by a SYS 7499. It is important to turn off the clock when you are not using it because some Vic features (such as the cassette unit) interfere with it and may cause a crash.

A little hint: When you have keyed in this program, save it before you run it! Unless every one of the numbers has been typed in correctly the Vic will crash and you may lose your program.

If you don't like the colour, change it — it's controlled by the 150 at the start of line 180. This just gives the number to be POKEd into the colour memory area at 38400.

Dear Tommy, Recently I bought a 16k RAM expansion pack for my Vic. Please can you tell me if there is any way in which I can load and run an 8K Vic program on my expanded Vic.

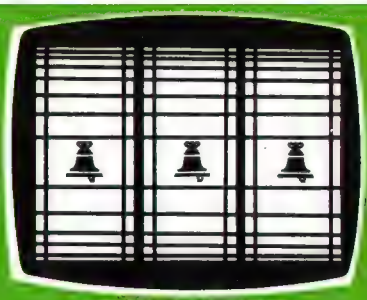
It does depend on whether the program is written in assembler or in Basic. If the former, then there is nothing you can do about it, I'm afraid — unless the supplier will exchange it for the correct version for your "new" Vic.

If the program is written in Baisc, though, there are some things you can try. As the Vic is expanded, some parts of the memory such as the screen area move around even the location of the Basic program itself changes. This means that any program which POKEs or PEEKs the screen memory will fall over if run on a Vic for which it was not intended.

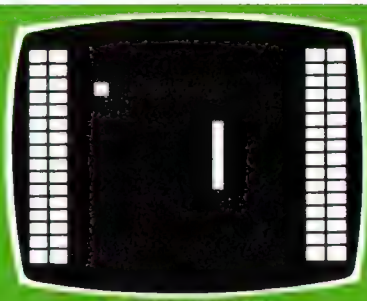
Some programs test to see if memory expansion has been fitted: but a lot do not I am afraid. If your program doesn't, you will have to go through it line by line changing any references to the screen and colour memory areas.

All you need to know now is what these addresses are, so here is a table to tell you just that:

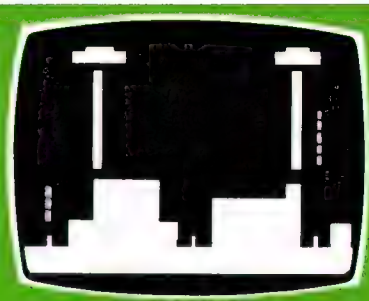
Size	Screen memory	Colour memory
5K	7680-8186	38400-38906
8K	7680-8186	38400-38906
Above 8K	4096-4602	37888-38394



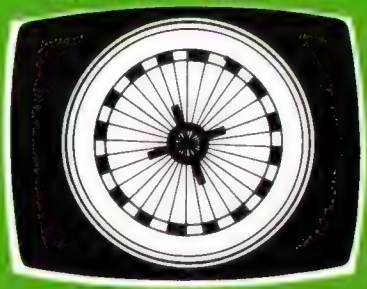
1. FRUIT MACHINE:



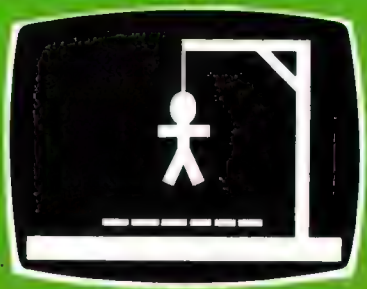
2. BRICKDOWN:



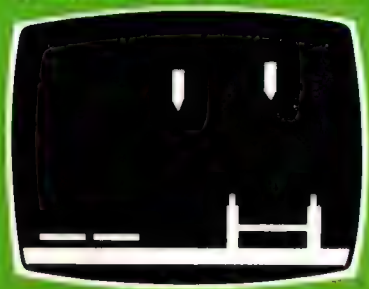
3. BLOCKADE:



4. ROULETTE:



5. HANGMAN:



6. TARKUS:

NEW!

only £9.95 inc. V.A.T. & Post & Packaging

BEELINES VIC 6 SIX VOL 1

The Games System in a Cassette!

At last! A really professionally produced piece of games software that will guarantee you hours of fascinating thrills on your VIC computer.

The first volume of the 'Beelines VIC 6' gives you exciting colour, arcade quality games that make full use of the power and display quality of the VIC.

The 'VIC 6' is attractively priced so that no VIC user need be without it, and it is available exclusively from Beelines. Just send £9.95.

So fill in the coupon, or phone us with your credit card number, and the 'VIC 6' will be yours within days!

BEELINES (Bolton) LTD
FREEPOST (No stamp required) Bolton BL3 6YZ
Ansaphone (0204) 385299
Reg. office: 124 Newport St., Bolton BL3 6AB.
Dealer enquiries welcome.

Please send me:

Qty.	Item	Price
	Beelines 'VIC 6' Vol. 1 at £9.95	

I enclose cheque/P.O. ☐
OR Please debit my:

Access 5224 ☐☐☐ ☐☐☐☐☐☐ ☐☐ ☐☐

Barclaycard 4929 ☐☐☐ ☐☐☐ ☐☐☐

Expiry Date _____

Name _____

Address _____

_____ Code _____

Tel. (day) _____

Official orders welcome –
we will ship to approved account holders on receipt of firm order.
Delivery free within mainland UK
Access and Barclaycard welcome.
Telephone answering machine for 24hr/7 day credit card orders **0204 385299**
Freepost: **Beelines, FREEPOST, Bolton BL3 6YZ**

VC/5/82

Butterfield

Program Transfers and Basic Merges by Jim Butterfield

Interest in the movement of programs between Vics and Pets continues to run at a high level among readers of Vic Computing. Jim Butterfield has been giving the matter some thought too...

If you have two Basic programs, it's hard to consolidate them together without getting typer's cramp. The command LOAD wipes the old program as it loads in the new one. This is a disappointment. There are times when you have a group of DATA statements in a program and would like to bring them in to another program which will use them for a new set of computations. The same thing is true of your favourite subroutines: it's annoying to have to type them in all over again.

Merging — true merging, that is — solves this for you. You can arrange to slip extra lines into your program as if you had typed them in at the keyboard.

Vic to Pet transfers

Merging can be used to transfer Basic programs from Vic to Pet. A merged program will occupy the proper memory addresses as it arrives into the Pet. LOADing a Vic program into the Pet often doesn't work.

There are potential problems in moving a program between Vic and Pet. For one thing, Vic colour won't show on a Pet.

If the program has PEEKs and POKEs, chances are it will take quite a bit of work to fit it into the other machine. If you're lucky, they can be changed to PEEK or POKE a new set of addresses; but it's not always possible to find a one-to-one translation between Vic and Pet.

If the program contains machine language — look for a SYS command or USR function — you'll probably have problems cutting it over to the new machine. Some machine language programs won't even work on all models of Pet — so a move to or from Vic would be much too big a shock. And the method that I will outline below won't work on machine language programs, anyway; just pure Basic.

Writing out the program

To transfer a Basic program, we're going to write cassette tape in an unusual way. It won't be a normal program tape: instead, it will be something called an ASCII listing tape. It will take about twice as long to write, and occupy about twice as much tape . . . but it will be compatible.

Here's show to write this type of tape. Type:

```
OPEN 1,1,1, "PROGNAME": CMD 1: LIST.
```

. . . and as soon as you press the RETURN key, you'll be requested to PRESS PLAY AND RECORD. Do it, and the tape will start. If you watch carefully, you may see the tape hesitating every few seconds or so. Eventually the tape will stop. When it does, type:

```
PRINT  1 : CLOSE 1
```

. . . and tape will move one last time. When it stops, the computer will say READY and you may rewind the tape and take it out of the drive. Basic programs flow easily from the Pet to the Vic. Vic has a relocating feature that repacks the incoming program into whatever space that Vic has available. That's a necessary trick, since Basic may start in any of three different locations within the Vic depending on what extra memory has been plugged in.

It's a little more difficult to transfer programs from the Vic to the Pet. The stable Pet system expects a Basic program to be set up for one specific start location. The Pet isn't equipped to handle this flighty VIC format which seems to hop around from place to place.

So for Vic-to-Pet transitions we need to cope with a memory mapping problem. There are several ways to approach this; some of my favourites are itemized below.

Keep in mind that we're discussing Basic programs only. Machine Language may need different and special handling. And don't forget: PEEKs, POKES, WAITs, USRs, and SYSs will probably need coding changes to work successfully in the new environment.

A few methods:

1. LOAD the program into the Vic with the 3K expansion module in place. SAVE the program. The program will load correctly into the Pet with no further modifications needed.

   Note that the Vic must have the 3K expansion and no other. The 8K expansion job won't work.

2. Before loading the Vic program into the Pet, type NEW; then FOR J=0 TO 2:POKE 4096+J,0: POKE 4608+J,0: NEXT J.

   Load the program into the Pet. Type LIST. If the program isn't there, type POKE 41,16. Type LIST. If the program still isn't there, type POKE 41,18.

   Don't forget to reset your Pet when you've finished playing with the Vic program.

Each of the above combinations corresponds to a Vic configuration at the time the program was saved. If 3K expansion was in place, the first LIST will work without any POKE (see method 1). If no memory expansion was there, POKE 41,16 will do the trick. If 8K or more was there, POKE 41,18 is he magic combination.

3. There's another technique available called a 'Merge'. Space is insufficient to outline the whole story here. Suffice it to say that you can use a complex piece of magic to perform a cassette tape merge; you can use special software for a disk merge (for example, POWER has this feature) or you can use a machine language program specially written to do the trick.

The capability of merging Basic lines together is quite important: its usefulness extends far beyond the simple objective of transferring programs between machines. We'll tackle the mechanisms another time . . .

You have some options on the above procedure. You may call the ASCII listing anything you like: instead of PROGNAME you can call it WHISKERS or CLOUD 9. It's a good idea to give a meaningful name to tape files; when you have 50 or more tapes sitting around you'll be very happy to get a hint as to what's on a given tape. You could if you wished write part of a program to tape instead of the whole thing: for example, you might type LIST 300-400 instead of just LIST in the first line.

You have quite a miraculous thing on the cassette tape. It's a program, but it's written as a data file. We could read the program as if it were data, analyze it, and do any kind of computing on it we wanted to. That's unusual: programs are programs and data is data — they seldom mix.

Getting ready to bring it back

When we recall the program from this oddly formatted tape, we will bring in the lines, one at a time, and merge them with any program already in place in the computer. It will work just as if we typed the lines. New lines will fit into the program in the correct line number sequence; and if a new line number matches an old one, the new line will replace the old one.

If we are just transporting a program from Vic to Pet, we must say NEW. This means that we are merging the program with nothing. The result will be the program by itself — but properly placed in the Pet.

If we want to merge the program we have saved with another program, now's the time to bring that other program into your computer. The lines from tape will mix in:

The Magic Merge

Don't try to understand it. Just do it carefully. In the following, PET4 is for 4.0 Machines, PETU is for Pet Upgrade Machines (they power up with "*** Commodore Basic***" on the screen and PET1 is for the Original ROM machines.

Put your 'merge' tape — the one we just wrote — into the cassette drive of the computer.

Now type:

VIC: POKE 19,1 : OPEN 1
PET4: POKE 16,1 : OPEN 1
PETU: POKE 13,1 : OPEN 1
PET1: POKE 3,1 : OPEN 1

. . . and when you press RETURN, you'll be asked to PRESS PLAY. Do it, and the tape will mvoe briefly and the computer will report FOUND PROGNAME.

We're almost there, but you must follow the next instructions very, very carefully. Clear the screen, and type exactly three cursor down characters. Watch it! The cursor-down key may repeat if you hold it too long. Type the following starting on line 4 (if you've followed instructions, you must be on line 4, right?):

VIC: PRINT "[home]": POKE 198,1: POKE 631,13: POKE 153,1
PET4: PRINT '[home]": POKE 158,1: POKE 623,13: POKE 175,1
PETU: PRINT "[home]" POKE 158,1: POKE 623,13: POKE 175,1
PET1: PRINT "[home]": POKER 525,1: POKE 527,13: POKE 611,1

The designation [home] above means 'press the home key'; the computer will print a reverse-S character. Don't type the letters H-O-M-E; that won't get you anywhere.

After you've input the above line, press RETURN and things will suddenly get very busy. The cassette tape will start to move, and it will keep moving with brief stops for some time. There will be no sign of activity on the screen, except that the word READY may mysteriously appear above the line you typed.

Eventually the tape will stop moving and an error notice will print. It might be ?OUT OF DATA and it might be ?SYNTAX ERROR — but in either case, ignore it: it's not a real error.

Your program is now in the machine. You may go ahead and use it, or SAVE it in the more conventional way for future use.

How it works

It's magic.

The basic procedure was evolved by Brad Templeton. If you want more details and happen to run across Brad, ask him: but you'd better have a week to spare. (Editor's note: Brad Templeton is an American, one of that freemasonry of Pet geniuses who seem to operate on a different intellectual plane to us mere mortals where Pet programming is concerned. He's probably best known for a fine Pet utility called POWER.)

Final remarks

You can merge programs together. You can transfer programs from Vic to Pet (or vice versa, for that matter). But we've only just begun to tap the treasures of the MERGE sequence.

For a few glorious moments, the tape unit took over control from the keyboard. Everything 'typed in' from tape was executed; it just happened to be program lines in this case. We have broken the distinction between data and program files, and a world of new possibilities e-merges. Programs that write program? Prgrams that control the computer's other activities? They are all possible.

audiogenic Ltd.

PROCEP CURSOR CREATIVE SOFTWARE PET PACK SOFTWARE commodore COMPUTER UMI

AMOK . . . The halls of AMOK are populated by robots out of control and out to get you. To save yourself you must be quick on the draw and fast on your feet. **VP010 *£6.99.***

SIMPLE SIMON . . . Puts your dexterity and memory to test. You have to follow a sequence of flashing colour bars and tones. **VP011 *£6.99.***

VICalc . . . Turns the VIC into an easy to use programmable calculator. Ten memories are displayed on the screen along with four working registers. Using single keystrokes all normal addition, multiplication etc. and scientific functions may be undertaken as well as compound interest and percentage. Precision may be set to round up the last digit from 0 to 9 decimal places. **VP012 *£8.99.***

A-MAZ-ING . . . Needs 3K expansion. Fast action gobbler game - eat the dots and avoid the nasty ghosts. You can eat the ghosts when they change colour. **VP016 *£6.99.***

MASTERWITS . . . Try to deduce the pattern of four markers from the six colours at your disposal, while Masterwits gives you the clues you need. Rumour has it that seven tries is average!? **VP023 *£6.99.***

KIDDIE CHECKERS . . . Teaches small children to play the game of draughts. Only looks one move ahead. **VP024 *£6.99.***

WALL STREET . . . The stock market comes to life. Follow the market's daily rise and fall with cash, shares and hope. Sharp trading will win. **VP025 *£6.99.***

ALIEN BLITZ . . . How good are you at blasting aliens from the sky? Find out on the superhuman ninth level. **VP026 *£7.99.***

SKYMATH . . . Needs 3K expansion. Specially for young children! Addition and subtraction problems set in a highly entertaining audio-visual presentation. The sound and visuals provide the incentive to learn. **VP029 *£6.99.***

SPACE DIVISION (Level 1) . . . Needs 3K expansion. Another one for kids! This program sets division questions in the context of a rocket countdown. As the questions are answered correctly, the rocket prepares for blast off, until finally the child is rewarded by seeing the rocket take off. **VP030 *£6.99.***

INVADER FALL . . . Another variation on the Invader theme. This time the unpredictable aliens fall from the sky at different points across the screen. You must shoot them before they touch ground. Accuracy and quick reflexes essential! **VP032 *£6.99.***

THE ALIEN . . . Needs 3K expansion. You are the Alien. You have landed in a maze inhabited by unfriendly creatures. You must trap the creatures in your inflatable traps to avoid being eaten by them. A totally original game concept! **VP033 *£7.99.***

STAR WARS . . . A game for all you budding Skywalkers! You are the space gunner fighting off the interceptor ships of the evil Empire. **VP034 *£6.99.***

HANGMAN-HANGMATH . . . HANGMAN . . . As it suggests. You must guess the word before you get hung from the gallows. HANGMATH . . . The same except that you have to solve a mathematical problem. **VP044 *£7.99.***

MATH HURDLER-MONSTER MAZE . . . MATH HURDLER . . . Is designed to teach basic maths. If you answer wrongly the hurdler crashes. MONSTER-MAZE . . . You are in a maze and have to escape by the exit without crashing into walls or being caught by the monster. **VP045 *£7.99.***

SEAWOLF-BOUNCE OUT-VIC TRAP . . . SEAWOLF . . . You are a submarine and you have to sink as many enemy ships as possible in 60 seconds. BOUNCE-OUT . . . Full colour 'Breakout' type game. Keep the ball in play while breaking down the wall. VIC TRAP . . . A battle between you and the VIC. Try to cross the screen whilst enticing the VIC to cross your path to score extra points. **VP046 *£8.99***

CODE MAKER-CODE BREAKER . . . CODE BREAKER . . . The VIC has to guess what code or pattern you have made. CODE MAKER . . . You have to guess the code or pattern that the VIC has produced. You are rewarded with black or white tokens as to the number of guesses you have made. **VP047 *£7.99.***

NEW **KOSMIC KAMIKAZE** . . . Requires 3K or 8K expansion. Destroy the suicidal aliens as they try to land on Earth. Avoid the deadly beams from the Mothership. A fast-action Invader Fall type game. **VP053 *£7.99***

NEW **MINIKIT** . . . The low-cost high power VIC toolkit program for any memory size! This cassette program automatically relocates itself at the top of memory, taking up about 1K. Using Minikit you can enter the common Basic Keywords with just one keystroke. Toolkit commands include Find, Kill, Delete, Auto and Trace. **VP054 *£7.99***

NEW **GOLF** . . . At last - Golf in the privacy of your own home! Set your handicap and tee off - with choice of clubs and nine different holes with hazards, bunkers, water, etc. About the only thing it doesn't do is serve drinks in the clubhouse afterwards! **VP055 *£7.99***

CARTRIDGE PACKS

SPIDERS OF MARS . . . The most incredible VIC game yet! You are a trapped fly and you must shoot your way across Mars avoiding the dreaded Spiders and the other flying creatures. With classical music! **VP014 *£24.99.***

NEW **CLOUDBURST** . . . Save the Earth from the downpour of Acid Raindrops and the invasion of the mutant Cloud Hoppers. Speed and skill are essential! With original music and ten levels of play. **VP048 *£19.99.***

NEW **RENAISSANCE** . . . The ultimate Othello package for the VIC! This age-old game has been brought right up to date. You can play against the VIC with eight playing levels, or use the VIC as the board to play another person. During the games you can change sides and playing level, take back moves, set up and play special games, and even save and recall games to and from tape! **VP049 *£24.99.***

NEW **SATELLITES AND METEORITES** . . . The amazing Asteroids type game cartridge for the VIC. All the facilities of the famous Arcade game, plus . . . It uses the full screen! No more borders - the final frontier overcome! **VP050 *£24.99.***

NEW **METEOR RUN**. . . Pilot your spaceship through the Meteor belt, blasting your way through the rocks and Alien saucers. A linear type of Asteroids game! Includes early warning radar display. **VP051 *£24.99.***

NEW **BUTI** . . . The VIC programmers utility cartridge. Commands include AUTO, DELETE, DUMP, EDIT, FIND, HELP, KILL, OFF, RENUMBER, REPEAT, STEP, TRACE, UNNEW, plus Hex to Decimal (and vice-versa) converter and special VIC command which reconfigures the memory. All this PLUS A FREE 3K MEMORY EXPANSION BOARD INCLUDED! How do we do it??? **VP052 *29.99***

Now also representing HES Human Engineered Software

VIXEL from THE CODE WORKS

Programs include Assembler-Editor, Text-Editor,
Pet Vic commnications package and yet more games.

DINERS AMERICAN EXPRESS VISA

ALL PRICES INCLUDE V.A.T.
AVAILABLE FROM ALL GOOD DEALERS OR DIRECT FROM

AUDIOGENIC, P.O. Box 88, Reading, Berks. Tel: Reading (0734) 586334

A matter of routines

Computers are often required to store or manipulate dates. Our calendar system is, unfortunately, an awkward system to use with its months of differing lengths and leap years. The programs given here overcome these problems by assigning simple numbers to each day. (The routines are based on algorithms suggested in Peter Duffet-Smith's book 'Practical Astronomy with your Calculator').

Date to Day Number

The routine DAY TO DAY NUMBER converts the months and days of a date to the number of that day within the year. For example, 23 November is the 327th day of the year — unless it is a leap year in which case it is the 328th. This could be stored as, say, 1980.328 or 1982.327. Line 110 checks for it being a leap year, correctly regarding a year such as 1900 as a non-leap year.

The day number is calculated in line 120 or 130.

The converse case of finding the date of a particular day number in a year is dealt with in the second routine, DAY NUMBER TO DATE.

DATE TO DAY NUMBER

ENTER ROUTINE WITH
- Y year
- M month
- D day

```
100 REM DATE TO DAY NO.
110 L=0 : IF Y/4=INT(Y/4)
    THEN L=1 : IF
    Y/100=INT(Y/100) AND
    Y/400<>INT(Y/400) THEN
    L=0
120 IF M>2 THEN
    DN=INT((M+1)*30.61)
    -63+L+D : GOTO 140
130 DN=31*(M-1)+D
140 RETURN
```

EXIT ROUTINE WITH
- DN day number,
- L=1 for leap year, otherwise L=0,
- Y,M,D unaltered.

Modified Julian Date

The Modified Julian Date is a continuous count of days beginning with day 0 on Wednesday 17 November 1858. The routine DATE TO MJD calculates this number for any given date. The resulting MJD is a single number that unambiguously identifies a particular day. It can thus be used as an easy way of storing date information. It is very convenient if the number of days between two dates is sought, it is only necessary to subtract their MJDs.

Dates before MJD 0 are, of course, given negative numbers. The routine will not, however, deal correctly for dates before 14 September 1752. This was the date in England for the change-over from the Julian to the Gregorian calendar: September 2 of that year was followed by September 14. There were no intervening dates. 1752 was also the first year to start on 1 January: previous years began on 25 March. Note that for accounting purposes the equivalent date in our present calendar, 6 April, is still the start of the financial year.

Routines are given to change a date to MJD and the converse.

DATE TO MJD

RETURN ROUTINE WITH
- Y year
- M month
- D day

```
100 REM DATE TO MJD
110 IF M<3 THEN Y=Y-1 :
    M=M+12
120 A=INT(Y/100)
130 B=2+A+INT(A/4)
140 C=INT(365.25*Y)
150 E=INT(30.001*(M+1))
160 MJD=B+C+D+E-679006
170 IF MJD<-38779 THEN
    PRINT "DATE BEFORE
    GREGORIAN CALENDAR
    IN ENGLAND"
180 RETURN
```

EXIT ROUTINE WITH
- Y,M,D unchanged
- MJD holding Modified Julian Date

OTHER VARIABLES USED IN ROUTINE
- A,B,C,E

Day of the Week

These routines can be used with the simple DAY OF WEEK routine to calculate to which day a date refers. It could, for example, be used as a check on data entry by ensuring that the name of the day is compatible with the date.

DAY OF WEEK

ENTER ROUTINE WITH
- MJD Modified Julian Date

```
100 REM DAY OF WEEK
110 W=MJD+3-7* INT
    ((MJD+3)/7
```

EXIT ROUTINE WITH
- W holding day of week, i.e. W=0 for Sunday,
- W=1 for Monday,
- W=2 for Tuesday etc.

DAY NUMBER TO DATE

ENTER ROUTINE WITH
- Y year
- DN day number

```
100 REM DAY NUMBER TO
    DATE
110 L=0 : IF Y/4=INT(Y/4)
    THEN L=1 : IF Y/100
    =INT(Y/100) AND Y/400 <>
    INT(Y/400) THEN L=0
120 IF DN >60+L THEN
    M=INT(DN/31.1) + 1 :
    D=DN-(M-1)*31 : GOTO
    140
130 M=(DN+63-L)/30.61 :
    D=INT((M-INT(M)*30-
    .61+1) : M=INT(M)
140 RETURN
```

EXIT ROUTINE WITH
- M month
- D day number
- DN, Y unaltered
- L=1 for leap year, otherwise L=0

Date of Easter

The final routine enables those who wish to plan ahead to find the date of Easter Sunday. The method used involves a series of divisions with the integer and fractional parts of the result being dealt with separately. This method was first published in 1876.

DATE OF EASTER

ENTER ROUTINE WITH
- Y Year

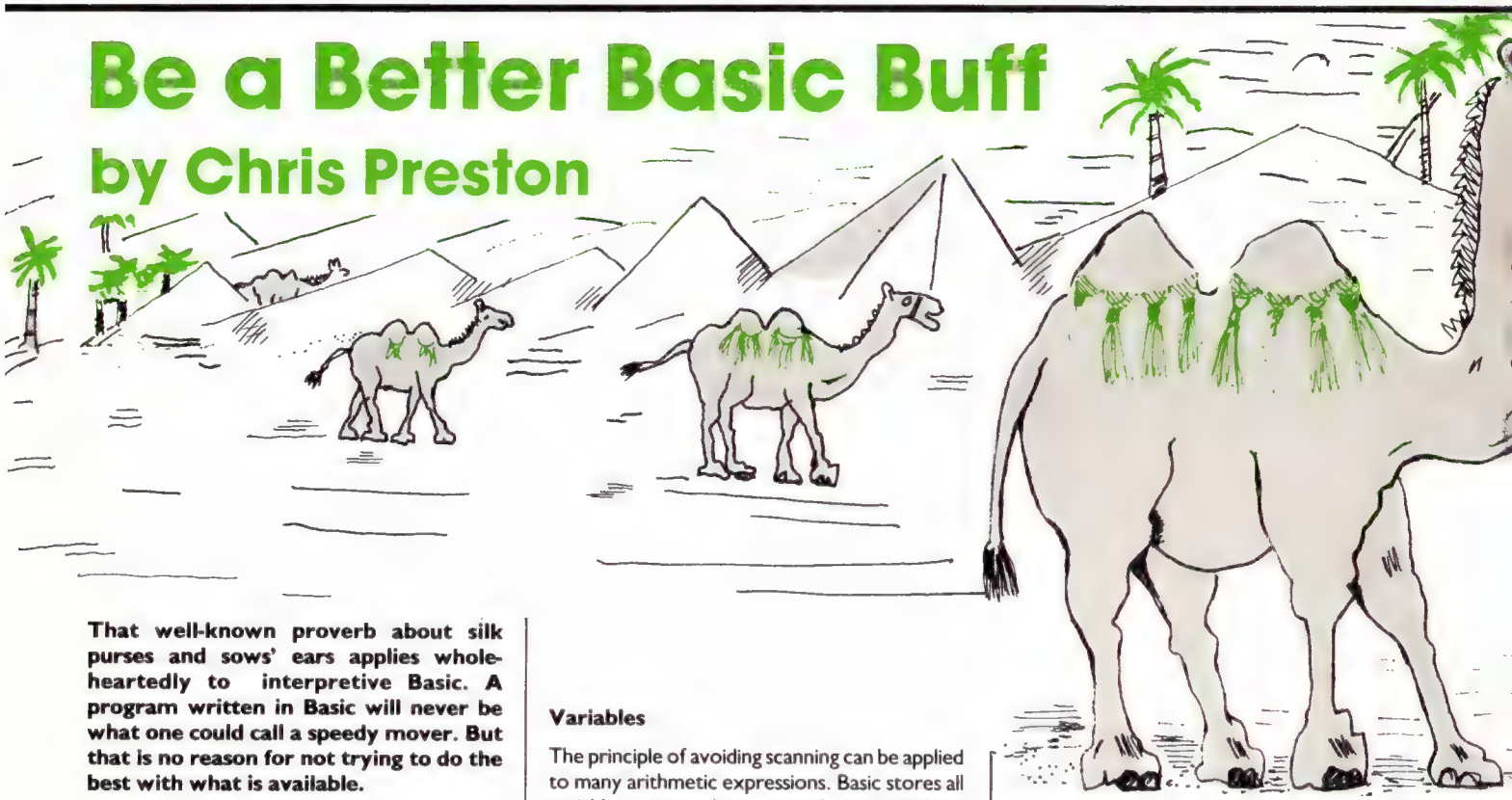
```
100 REM DATE OF EASTER
110 A=Y-19* INT(Y/19)
120 B=INT(Y/100) :
    C=Y-100*B
130 D=INT(B/4) : E=B-4*D
140 F=INT((B+B)/25)
150 G=INT((B-F+1)/3)
160 H=(19*A+B-D-G+15)
170 H=30*(H-INT(H))
180 I=INT(C/Z§) ,
190 K=C-4*1
200 L=(32+2*E+2*I-H-K)/7
210 L=7*(L-INT(L))
220 M=INT((A+11*H+22*-
    L)/451)
230 N=INT(H+L-7*M+1-
    14)/31)
240 P=H+L7*M+115-31*N
250 IF N=3 THEN
    M0$="MARCH"
260 IF N=4 THEN
    M0$="APRIL"
```

EXIT ROUTINE WITH
- M0$ month
- P the day of Easter Sunday
- Y unchanged

OTHER VARIABLES USED IN ROUTINE
- A,B,C,D,E,F,G,H,I,K,L,M,N.

Be a Better Basic Buff

by Chris Preston



That well-known proverb about silk purses and sows' ears applies wholeheartedly to interpretive Basic. A program written in Basic will never be what one could call a speedy mover. But that is no reason for not trying to do the best with what is available.

There are many publications and courses that will teach you how to program in Basic, and even more are available on programming and analysis techniques in general. But you will find very little in the way of advice on how to make programs run faster, because those books and courses tend to concentrate on abstract ideas — either a language study, for example, or a theoretical look at methodology.

This article aims to give a little practical advice on how to tweak your programs to make them run swifter, or how to squeeze a few extra bytes of program into memory.

The most important thing to remember is that Basic as implemented on the Vic (and on most other microcomputers, in fact) is an **interpretive** language. This means that large parts of the total execution time for a program will be spent by the Basic in scanning the program text. For instance, when Basic executes a GOTO or GOSUB instruction, it has to search for the line number, referred to by the instruction. If this is greater than the current line number, the search starts from the current line; if it is less, the search starts at the beginning of the program.

For that reason, if the program is at all large any commonly-used subroutines should be placed at the start of the program to cut down on this searching time. In a pretty typical medium-sized 8K program, a GOSUB to the end of the program took 15.4 milliseconds: a GOSUB to the start of the program took a mere 1.1ms.

Variables

The principle of avoiding scanning can be applied to many arithmetic expressions. Basic stores all variable names and numeric values in a table at the end of the program, where they appear in the order in which they were encountered in the program. When Basic wants to get the value of a variable, it has to search down the variables in this table one by one until it finds the one it wants. If the variable is some way down the list this can take some time (arrays are stored separately, so A(0) to A(1000) only counts as one name).

For example, the expression A = 1 is evaluated in 1.45ms when A is the first variable in the list. But if A is the twentieth variable the assignment takes 2.6ms, and large program will contain many more than 20 variables.

it makes sense to make sure that frequently-referenced variables, especially FOR-variables, are placed at the start of the list. This can be done by some dummy assignments at the start of the program. The difference in the time taken by Basic to parse a short variable name (like 'T') and a longer name (such as 'TEMPERATURE') is insignificant.

Constants

if your program contains a large number of constants, such as 123.456, bear in mind that each time 123.456 occurs in the programs it takes up seven bytes; and that the assignment A = 123.456 takes 14ms to execute, the extra time being taken up in converting the character string "123.456" into floating point so that it can be assigned to A.

It is far better to say Q = 123.456 at the start of the program and replace every occurrence of 123.456 in the program with Q. This will save six bytes and about 13ms every time!

On the other hand 'short' constants such as 1 and 5 may actually be slightly faster than a variable reference, as Basic does not have to search the variable to find the value.

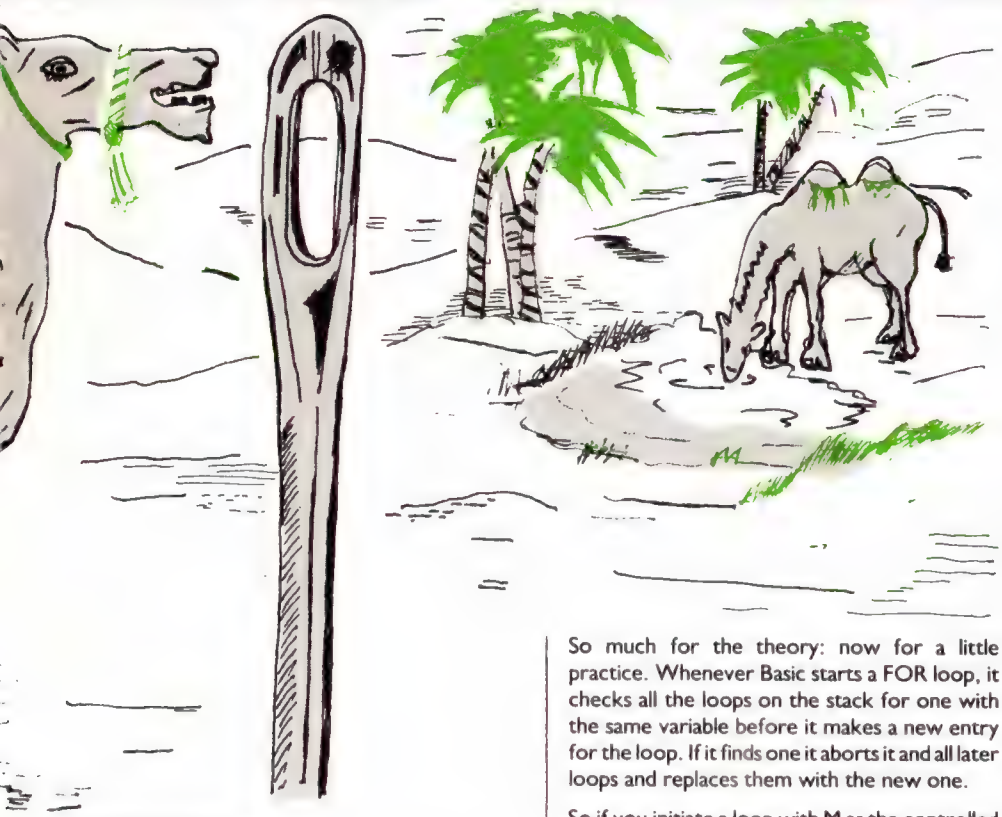
While on the subject of flating point arthmetic, it's worth noting that the exponentiation operator, for instance A3 (A to the power of 3) is appallingly slow. Unless you want to raise a number to a variable power (A V), it is much faster to use repeated multiplication: A*A is 10 times faster than A2.

Also beware of the trigonometric functions such as SIN and COS. These operate by evaluating a power series, and so are not very swift.

Integer Variables

One of the features of Basic is the use of integer variables, or so the standard manuals would have it. Don't believe it! Vic Basic cannot do integer arithmetic; and integer variables take up the same amount of space as real variables, which means five bytes for the value and two for the name. (Actually, that is not quite true. Each element of an integer array only uses two bytes, so you can save space using integer arrays rather than real arrays where possible.) If Basic is faced with the expressioin A% = B% + C%, it first of all searches for B%, converts it to floating point, searches for C%, converts that to floating point, adds the two values and converts the result to integer ready for the assignment.

Simple integer variables are thus a waste of time (literally as well as metaphorically). The only reason they are provided is to allow compatibility with versions of Basic on other machines.



FOR-NEXT Loops

It is obviously important to keep delays inside a loop as small as possible, because every tiny inefficiency is multiplied by the number of times the loop is repeated.

Before going into ways to make your FOR-loops faster, here is a good place to explain something I know puzzles many people — the rules that control exactly when you can jump out of FOR loops before they finish. If you do this haphazardly, you can end up with very strange results from your program. The rules are quite easy once the principle of how Basic controls the execution of loops has been understood.

We first of all have to describe a construction much used in programming, especially assembly programming. a **stack** is a structure which has the property that when you take something out of it, what you get is the last thing you put in. This effect is sometimes referred to as LIFO (Last In First Out), and a stack is sometimes called a 'pushdown store'.

Stacks can be programmed easily enough using an array and a pointer variable, and the stack used by Basic to control FOR loops and GOSUBs amongst other things is situated in memory starting at 511 and going down to 256. The size of this stack poses an obvious limitation int he maximum number of nested loops or GOSUBs you can have.

For every FOR loop which is currently active, there will be an entry on the stack giving the FOR variable, the 'TO' value, the 'STEP' value and the address of the start of the loop. When a FOR loop terminates, its entry is deleted from the stack.

So much for the theory: now for a little practice. Whenever Basic starts a FOR loop, it checks all the loops on the stack for one with the same variable before it makes a new entry for the loop. If it finds one it aborts it and all later loops and replaces them with the new one.

So if you initiate a loop with M as the controlled variable, then within that loop start another loop with J, the J entry will be below the M entry on the stack. As the J loop should finish before the M loop, this is no problem; the structure is ideally suited to control nested loops. If you now try to nest another M loop, however, havoc will break out. Basic will abort the existing M and J loops, so when the new M loop finishes and the NEXT J is reached you will get a NEXT WITHOUT FOR error.

The moral is to use different names for different loops in the same nest, and to be careful when jumping out of active nests. This should be done as follows:

```
10 FOR K=1 TO 10
20 FOR M=1 TO 10
30 IF A(M,K)=0 THEN M=10: GOTO 50
40 NEXT K
50 NEXT M
```

After that aside, back to the business in hand. There is actually not a lot that can be done to speed up loops. All the interpretation is performed by the FOR statement; the NEXT statement is provided with the TO and STEP values, the address of the FOR variable (no searching the variable table), and the address of the start of the loop (no searching for the line number of the start of the loop) it is thus quite efficient.

But specifying a variable after a NEXT, as in NEXT I, will slow down the loop by about 25 per cent. The other thing to watch is that if you are nesting two loops, try to order them so that the inner loop performs the higher number of iterations. Look at the timings for these two loops:

```
100 FOR M=1 to 10
110 FOR J=1 to 1000
120 NEXT : NEXT
```

. . . 9.02 seconds

```
100 FOR M=1 to 1000
110 FOR J=1 to 10
120 NEXT : NEXT
```

. . . 13.18 seconds

The reason for the difference is that there is a time cost associated with a FOR statement as with any other. But in the first example the second FOR statement is only executed 10 times; in the second it is executed 1,000 times.

As far as strings are concerned, the great bugbear is the question of 'garbage collection'. The area between the end of the program and the top of memory is used to store string values manipulated by the program. String constants, those appearing in the program inside quotes, are held in the program itself and do not form part of the problem. When a string, A$, is given a value, the string data is copied to the top of memory and assigned to A$. If another variable B$ is given a value, its data will appear below that of A$. When A$ is given a new value, this is also copied to the bottom of the string data: and although the old data for A$ is now redundant, 'garbage', it remains in position, taking up space.

In this way the string data fills up memory from the top, and eventually, this reaches the top of the program,. When this happens, a 'garbage collection' routine is called by Basic to throw away all the garbage and push down all the valid data. While this routine is running, the Vic will become completely dead.

The length of time taken for a single garbage collection depends only on the number of strings used by the program. A larger program means that the collection routine is called more often, as there is less memory available for dynamic strings.

The important thing to notice is that this is a linear relation, whereas the time taken per collection depends upon the square of the number of strings. In this way splitting up a large program (which is a good thing anyway, as several small modules are easier to debug) can also improve execution time.

it is obviously important to minimise the time spent collecting garbage. The biggest contribution to this is made by large string arrays. If these can be avoided, so much the better.

Another factor is the amount of string handling performed by the program. Look at the following loop:

```
100 FOR M=1 TO 212 : A$=A$+" " :
NEXT
```

This innocent looking loop appeared in a program from a quite reputable software house (though it was not a Vic program). It generates about 22,000 bytes of garbage and took around three seconds to execute.

computer world®

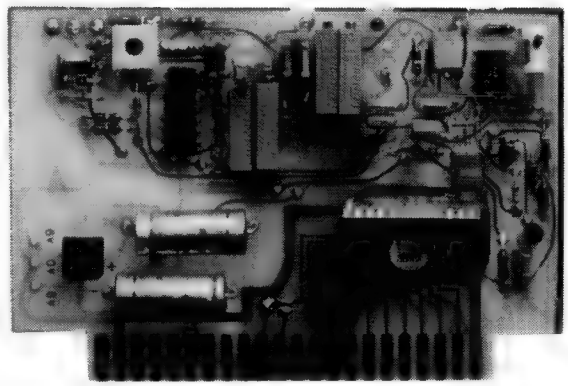
Division of Radcom Electronics

See us on the Arfon stand at the Commodore Computer Show on June 3rd 4th 5th Cunard Hotel Hammersmith

TELEX CW ON YOUR VIC

£89.00 (excluding VAT)

Plug-in module for standard VIC20. Comes with RTTY converter and extensive software in ROM. You only need short-wave radio or transmitter.



Specification
Input audio 100 m V
Output TLL 5V
to user port VIC20

CW transmit and receive adjustable speed from 6-36 wpm
Auto CW IDD every ten minutes

PTT SWITCH
RTTY transmit and receive baud rate 45, 50, 57, 75, 110 and 300.

TONES 1275, 1445, 2125 and 2125, 2250 2975 Hz

SHIFT 170, 425, 850 Hz

CW ID every ten minutes.

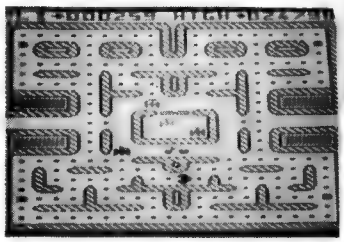
There are individual buffers of maximum 250 characters to store your own text. Five prepared buffers with the quick brown fox, station description CQ message etc.

Out of start memory to store text in your VIC from other transmitters.

We accept cheques/PO's payable to Computerworld. You may also pay by credit card. If so ring Netherlands 35 126 33 for immediate service. We accept Mastercharge and Visa. Price excludes VAT and shipping. Please add P&P at £5.00 per order and 18% VAT. **Computerworld, Hilvertsweg 99 1214 JB Hilversum, Netherlands**

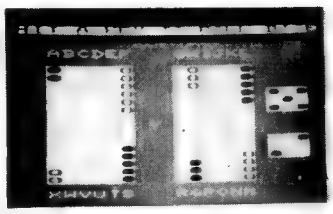
AT LAST

VIC SOFTWARE



VICMEN

A full-feature version of the popular "Puckman" arcade game for the UNEXPANDED VIC. Written entirely in machine-code for fast action. SUPPLIED ON CASSETTE @ £7.00



VICGAMMON

Standard Backgammon game for the VIC with 3K expansion. Rapid computer responses. Instructions on how to play are included. SUPPLIED ON CASSETTE @ £7.00

More VIC software will be available by the time you read this. Phone 051 227 2642 for details. generous DEALER DISCOUNTS available — phone Dave on 051 227 2299 for details.

ALL PRICES INCLUSIVE

98-100 THE ALBANY,
OLD HALL STREET,
LIVERPOOL L3 9EP

BARCLAYCARD VISA

BUG-BYTE

PCS

PEDRO COMPUTER SERVICES

VIC – 20 SOFTWARE (Basic)

BIG FOUR: (3.5K) Price £5.00
Meteorites: You have materialised in a meteor field and have to shoot your way out. The more meteorites you hit the more you will score.
Minefield: Use your skill to determine where the mines are burried, but you have to be quick as your mine detector is fading fast.
Hunter: A fast moving game in which you have to manoeuvre your man to hit sheeps on the screen. Beware of the many holes!
Lander: After a simulated take-off to approx. 800,000 m you have to control your thrust in order to land your ship safely.

STARTREK: 3K & 8K (& above) version Price: £5.00
You have to warp your way round the galaxy destroying Klingons. Your navigational computation always tells you where to find them. Using both plasm bolts and visual combat lasers, you must try to rid the galaxy of Klingons.

DATA BASE: (Retrieval System) for 8K and above Price: £7.00
Catalogue your records, stamps, books, etc. Define up to 7 fields.

HI–RES CHARACTER GENERATOR: 8K and above Price: £7.00
Created your own custom character set and save them on tape to be used with your own software. No number work required. Very easy editing features.

MACHINE CODE SOFTWARE *(very fast)*

THE CYBER MEN: 3.5K Price: £7.00
See how fast you are at eliminating the Cyber Men with your mines or see how many points you can score while you try to evade them.

VIC DEFENDERS: 3.5K Price: £7.00
Fly at zero altitude, dodge ground to air missiles, destroy fighter ship, super graphics.

HARDWARE

Graphic Tablet (high resolution)	Price: £90.95
Telephone Dialler	Price: £12.50
Light Pen	Price: £19.50
8-channel A/D Converter	Price: £75.00
6-channel Relay Switching Unit	Price: £45.00

Please add 15% VAT and £1.50 p&p (p&p for hardware only) to above prices.

Send s.a.e. for information on all our other products.

4 Cowcross Street London EC1 Telephone: 01-250 1481

VAT Reg. No. 371 1139 79

PROPRIETOR: R.I. PEDRO, Registered in England No: 2508331. Registered Office: 41 Brockley Rise, London SE23

Todd's Lore

by Mike Todd

Thirty-year-old Mike Todd hs been programming computers since he was 15 when his school acquired a Ferranti Pegasus computer — not an easy machine to program, but Mike soon learnt to write in machine code. About the same time Mike became licensed as a Radio Amateur (G8AZY) and is actively involved in hospital radio.

When he left school Mike joined the BBC as a television cameraman: he also started work on a degree course which included science, technology and maths. In 1973 he transferred to BBC Radio where he is currently a Technical Operator. As part of his job he can often be seen as the engineer "the other side of the glass" on many radio shows including Jimmy Young, Terry Wogan, and Dave Lee Travis (he has been known to do some spare-time computer work for some of the disc jockeys).

When the Commodore Pet was announced, Mike Todd was one of the first to order one. It arrived on 13th March 1978 at 1:13 pm (1313 on the 24 hour clock). If this wasn't coincidence enough, he was living in London W13 at the time and the Pet was serial number 1000013! Mike was soon writing the odd contribution for Commodore News and other magazines; and when he joined the Independent Pet Users Group (IPUG) he became a regular contributor.

As IPUG (now the Independent Commodore Product Users Group) started to get involved in the Vic, Mike took on the job of co-ordinating Vic activities. Since then he has become a regular contributor to Vic Computing as well as continuing to contribute to the ICPUG newsletter.

His field of interest is very wide: he has been quoted as being probably the most knowledgable person in the UK on the Vic, although he is equally knowledgable on the Pet and disks.

To do all this, it is perhaps not surprising that Mike is still single and lives on his own. His other interests include listening to music of all kinds (more especially rock), playing the piano, amateur radio, playing squash and going out to the pub.

Mike has recently been taken off his normal work and is currently working on a couple of projects involving computers and displays. As well as having to learn Pascal (to run on a Z80-based Zilong MCZ-2) he also spends some time on the office Apple equipped with disks, printer, graphics tablet and plotter.

Mike's own equipment includes his original 8K Pet, a 3032, 3040 disk drive, 2023 printer and a Vic-20. His ambition? To get away from office work and to get back to his shift work.

A common complaint I receive is that the video lead supplied with the Vic modulator is only a few inches long. It is possible to make up a new lead with a phono plug at one end and an aerial plug at the other, but this could be a fiddly task. The best solution I can offer is to get an aerial extension cable from your local TV shop which must have a socket at one end and a plug at the other.

The need to swap over Vic and aerial leads can be avoided by using a combining unit which takes two aerial plugs and runs them into one plug for the TV set. This will mean that using the Vic is simply a matter of selecting the correct channel. These combining units vary in price from £1.50 up to around £4.00 and are usually sold as 'aerial splitters' since they are originally designed for feeding two TV sets from one aerial.

If you don't live in strong signal area, this solution will unfortunately attenuate the signals and worsen your TV picture — your TV dealer should be able to advise on all these matters.

Because of the shortness of the video lead, many people are operating the Vic close to the TV set. If the cassette unit is also close it is possible for it to pick up interference from the TV. If you are getting inexplicable errors when using the cassette unit, this may be the cause: so try moving the unit up to a couple of feet from the TV.

Some of you may have had a problem with the POKE/SYS that I gave for positioning the cursor on the screen. The first problem arises if you use the command straight from the keyboard and not within a program. As soon as you hit RETURN, the cursor position is reset to the beginning of the next line — in this case put the PRINT statement on the same line (after a colon of course).

There is a possibility, however, that the SYS command won't actually work. This is because a third location should be preset. The complete sequence is now:

POKE 781, line: POKE 782, position: POKE 783,0 : SYS 65520.

The reason for the POKE 783,0 is to ensure that the status register has the carry bit set to zero. By using POKE 783,1 followed by SYS65520, the locations 781 and 782 will have the current line and position of the cursor.

The TI clock built into the Vic has caused problems for some. Unfortunately the Vic manual is not very clear on how it works.

Sixty times a second, the variable TI has 1 added to it. This starts as soon as you switch the Vic on and continues at all times. Therefore, if after a minute has elapsed you type PRINT TI, you should get something in the region of 3600.

There is also a variable TI$ which is updated once a second and which is in the form "HHMMSS" (that is, Hours, Minutes and Seconds) and reverts to 000000 after 235959. In the above PRINT command, had you typed PRINT TI$ instead you should get 000100 indicating one minute.

The variables TI and TI$ are just like normal variables that you would use in a Basic program — except that you cannot alter TI using TI = X. You can, however, alter TI$ by typing something like TI$ = "132000" which will set TI$ and TI accordingly as soon as the statement is executed.

A simple example of the use of the TI variables is to put a clock in the top left-hand corner of the screen. Type TI$ = "132058" (or whatever time is approaching) and at that time hit the RETURN key. TI$ will now stay at the correct time. You can then enter the following simple program:

```
10 PRINT TI$ "<home>
20 GOTO 10
```

<home> indicates that the HOME key should be pressed and this shows as a heart on the screen. When RUN is then typed, the correct time is displayed on the screen.

TI can be used to time events by setting it to zero (use TI$ = "000000"). Subsequently the time elapsed (in 1/60 second) can be found at any time by using TI in your program. It can also be used as a "time-out" timer for the end of a game, as follows:

```
10 TI$ = "000000"
110 IF TI$ > "000500" THEN STOP
```

Line 10 initialises the timer: line 110 can occur at any time during your program and in this example will STOP after 5 minutes. Of course IF TI > 1800 THEN STOP will have the same effect in line 110.

There is a problem with TI and TI$ whenever the cassette is being used since TI and TI$ use the same timer mechanism used by the cassette routines. If you do use the cassette you will find that TI and TI$ start counting far faster than they should. Unfortunately, there's not a lot one can do about it.

I've recently had an opportunity to try out a couple of Vic games cartridges which should be available very soon. The first is RAT RACE which is yet another arcade-style game. You are a mouse caught in a maze and your objective is to eat all 10 cheeses dotted around, avoiding the stationary cats and before the chasing rats catch you. Points are scored for each cheese eaten. Every complete run scores bonus points, but there are extra rats and cats for following screens.

It is, like all arcade games, addictive: and I found it good fun although I've still not scored more than 20000 and gained my extra life. Since you only see a portion of the maze on the screen, life can get very difficult; but there is a small mimic display at the side showing the position of you, the cheeses and the rats, just like a small radar display. If all this isn't enough, there is an irritating tune playing a line from "Three Blind Mice" throughout. It all adds up to another good, addictive game from Commodore.

The other game I tried is a Vic version of one of the most popular chess programs around, SARGON II. This uses Vic graphics well, although I found the letterings around the board difficult to read as a result of the choice of colours. But there is a facility to change these colours when you start.

Toolkit Lore

You may already have noticed that many of the notes produced by the Vic are not quite in tune. This is because of the method of deriving the tones on a simple division from the main oscillator.

The following formula will allow you to calculate the acual frequency generated by each of the three tone generators. R is the register (0,1 or 2), C the clock frequency of 1108404.5 Hz and N is the POKEd value. Freq=(C/21(8-R))/(255-N) note that if N is 255 then it should be replaced by -1. If N < 128 then there's no sound. This is quite a simple application for the Vic since a simple program could be written to calculate the frequency as follows:

```
10 C=1108404.5
20 INPUT "WHICH REGISTER"; R
30 IF R<0 OR R>3 THEN PRINT
   "ILLEGAL REGISTER": GOTO20
40 INPUT "WHAT POKE VALUE"; N
50 IF N<0 OR N>255 THEN
   "ILLEGAL VALUE" : GOTO40
60 IF N<128 THEN PRINT "NO
   SOUND": GOTO20
70 IF N=255 THEN N=-1
80 PRINT "FREQUENCY=";
   (C/21(8-R))/(255-N)
90 GOTO 20
```

This could easily be modified to actually play the note at the same time as displaying it s frequency.

Options exists to play a normal game, or to set up a specific position, You can play either black or white at seven different levels of difficulty. At level 0, response is almost immediate; but at level 6 the Vic's moves could take several hours — and at all times the Vic gives you information as to which moves it is considering. I'm not a chess expert but found that, at level 2 I was being given a good game. I've not had the time yet to play a game at level 6!

Moves are entered via the keyboard (in coordinate form) or using a joystick and the screen lists the last few moves. There's even an option to ask the Vic what move it recommends — but beware, you take its advice at your own risk!

I enjoyed playing SARGON II and feel that it could give most amateur chess players a reasonable run for their money.

SARGON II makes a welcome change from arcade style games, and if you are interested in these more "intellectual" style games, there is a new series of ADVENTURE games in cartridge form to be released soon written by Scott-Adams. As well as these, there will be a range of business/home utilities, and education rograms including GCE/CSE revision programs. If you thought that all the Vic could do was play games then you're in for a very pleasant surprise.

In addition, if you think that nearly twenty pounds is a lot to pay for a games cartridge (and if you don't, I do!) there should also be a series of cassette-based games from Commodore around soon. The first one is called BLITZ and is about a fiver. The objective is to flatten a skyscraper city from a slowly descending plane. The plane moves left to right and gets successively lower, so you have to raze the higher skyscrapers before you crash into them! It's an entertaining game with a reasonable amount of skill. It is also written in Basic so it may be an interesting exercise to try to analyse how it works and use the techniques for writing your own games.

Finally, most games cartridges start by allowing you to position the screen using the cursor control keys. Although this is usually done in machine code, you may like to incorporate this facility into your Basic programs; and Ian Logan from Lincolnshire has sent the following code which could be put at the beginning of your program. Note that the STOP in line 20 is executed when the "F1" key is pressed and shoud be replaced by a GOTO which will start the main program.

```
10 LK=197 : SH=653
20 IF PEEK(LK)=39 THEN STOP
30 IF PEEK(SH) THEN 70
40 POKE 36864,255 AND PEEK
   (36864)-(PEEK(LK)=23)
50 POKE 36865,255 AND
   PEEK(36865)-(PEEK(LK)=31)
60 GOTO 20
70 POKE 36864,255 AND PEEK
   (36864)+(PEEK(LK)=23)
80 POKE 36865,255 AND PEEK
   (36865)+(PEEK(LK)=31)
90 GOTO 20
```

"A leading computer company proves that if you've got something worth showing, it's good business sense to make an exhibition of yourself."

THE THIRD INTERNATIONAL COMMODORE COMPUTER SHOW OPENS JUNE 3RD

The Commodore Show is one of the finest opportunities to see the best of today's micro-computer systems in action. A wide range that covers our home colour computer and our selection of sophisticated business systems as well as the latest in software and related products.

Also we'll demonstrate just how they can improve efficiency, whatever line of business you're in.

There'll be seminars on education, communications and a wide variety of business applications.

Guests include knowledgeable and interesting people like Jim Butterfield, who's the foremost authority on the PET and its capabilities.

All in all, it's the biggest and best Commodore Show yet, and definitely not to be missed.

See us at:
The Third International Commodore Computer Show, Cunard Hotel, Hammersmith, London.

Thursday June 3rd	12am - 6pm
Friday June 4th	10am - 6pm
Saturday June 5th	10am - 5pm

For further details ring
SLOUGH (0753) 79292
675 Ajax Ave, Slough, Berks.

commodore COMPUTER

Quite simply, you benefit from our experience

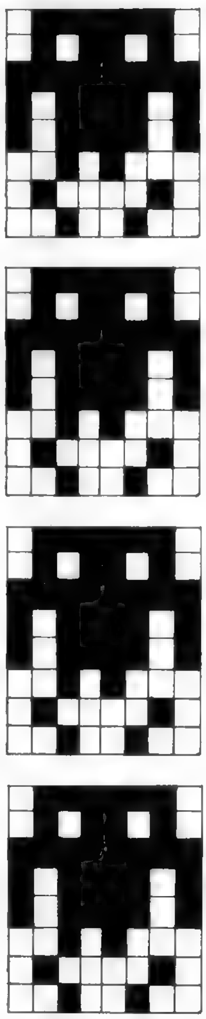
HOW THE VIC DISPLAY CONTROLS APPEAR IN LISTINGS PRINTED ON THE 1515:

```
CLR ...           (REVERSED HEART)
HOME ...          (REVERSED S)
RVS ON ...        (REVERSED R)
RVS OFF ...       (REVERSED UNDERSCORE)

CURSOR UP ...     (REVERSED SHIFTED Q)
CURSOR DOWN...    (REVERSED Q)
CURSOR LEFT...    (REVERSED UPWARD BAR - SHIFTED H)
CURSOR RIGHT..    (REVERSED LEFT SQUARE BRACKET)

SET COLOUR TO:
    BLACK ...     (REVERSED SHIFTED P)
    WHITE ...     (REVERSED E)
    RED ..        (REVERSED £)
    CYAN ...      (REVERSED COMMODORE-SHIFTED *)
    PURPLE...     (REVERSED COMMODORE-SHIFTED -)
    GREEN ...     (REVERSED UP ARROW)
    BLUE ...      (REVERSED LEFT ARROW)
    YELLOW...     (REVERSED PI SIGN)
```

```
5 REM ** FIRST BITS**
10 POKE650,128
15 DIMA%(4):SC=0
20 TI$="000000"
25 GOSUB12000
30 REM **MAIN BLOCK **
40 GOSUB5000
50 GOSUB4000
60 GOSUB6000
70 GOSUB7000
80 GOSUB8000
82 REM **CHECK FOR ALIENS ***
85 IFNA=0THEN11000
90 GOTO70
3998 END
3999 REM ***SUBROUTINES ***
4000 REM **INITIALISE**
4010 FU=20:FU$="■■●●●●●●●●●●●●●●●●●●●●"
4020 DIMI,J:SP$="■                    "
4025 DI=1:CO=0:NA=20
4030 POKE36878,8
4040 FORI=1TO4:A%(I)=3:NEXT
4050 LM$="■■■■■■■■■■■■■■■■■■■■■■"
4060 RETURN
5000 REM** GRAPHICS**
5005 POKE56,28:POKE52,28
5010 FORI=7168TO7168+55
5020 READD:POKEI,D:NEXT
5030 DATA126,90,255,189,189,36,66,36
5040 DATA36,126,219,219,219,126,36,195
5042 DATA24,24,24,24,24,24,90,219
5044 DATA7,15,31,31,31,63,127,255
```



Invasion

This modest interpretation of 'Space Invaders' gets into Victuals not so much because it's a great game, more because it exhibits some clever graphics programming. As a game it is deficient: it's more like a shooting gallery than an alien invasion, since the nsties don't get any lower and they don't fight back — as you pot, them with your (somewhat sluggish) laser gun (laser bow and arrow?) you're competing only your diminishing fuel supply.

That's shown as a line of red blobs at the top of the screen, and as you fire they will diminish (not quickly enough, in our opinion). If you run out, nothing much happens except that you get our score and the chance to try again.

If you succeed in knocking out all the aliens you get a time for your performance. The best we could do in the test runs was just over a minute: not too great! Part of the problem is that some of the Fiends From Outer Space are so light that they don't show on our Sony's screen: a neat trick . . .

Controls are good:

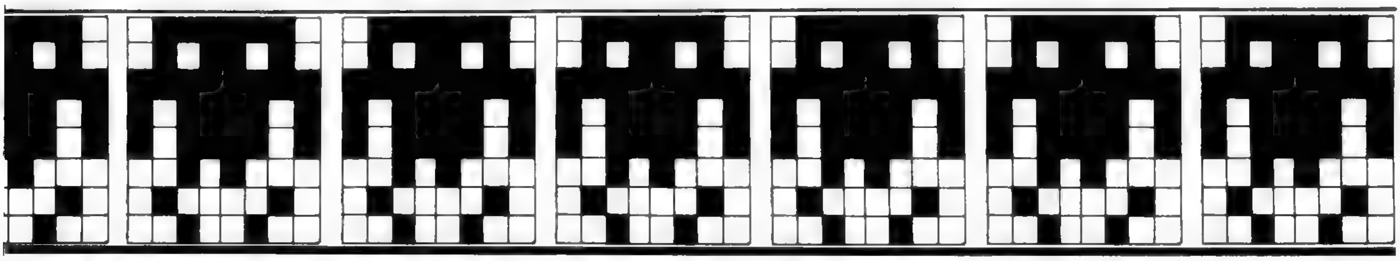
- P — left movement
- @ — fire
- ★ — right movement

That means you can play with one hand, thus leaving the other free for Frequent Libations.

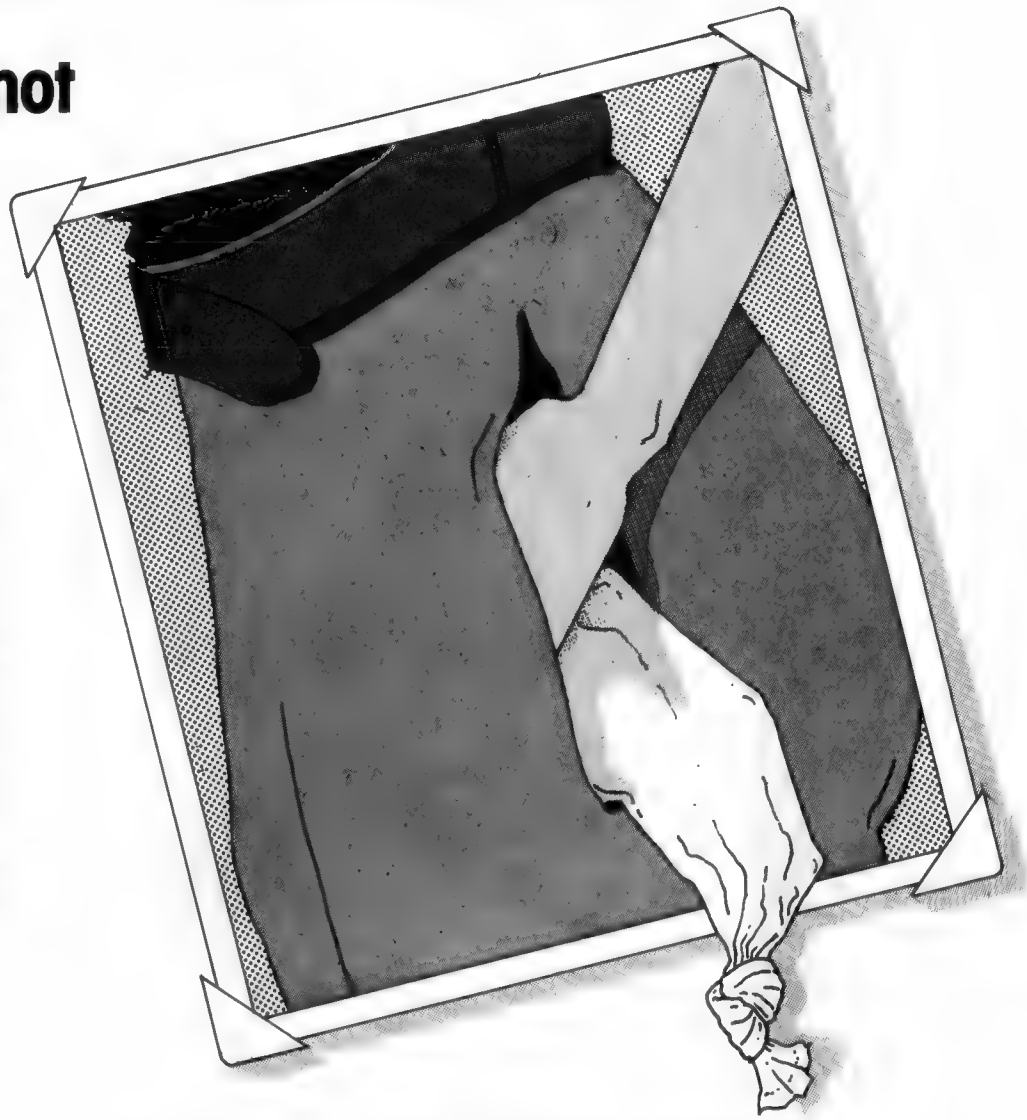
```
5046 DATA219,219,219,219,255,255,255,255
5048 DATA224,240,248,248,248,252,254,255
5049 DATA0,8,8,8,8,8,8,0
5050 DATA0,0,255,255,255,255,0,0
5055 FORI=7424TO7431:POKEI,0:NEXT
5060 POKE36879,238
5070 PRINT"▒▒";:POKE36869,255
5080 RETURN
5999 REM **SET UP SCREEN **
6000 AM$="▒▒▒▒▒"
6005 PRINT"▒▒▒●●●●●●●●●●●●●●●●●●●●●●  ";
6007 PRINT"▒▒────────────────────────";
6010 PRINT"▒";AM$:PRINT
6020 PRINT"▒    @  @  @  @  @"
6030 PRINT"▒▒    A  A  A  A  A"
6040 PRINT"▒▒     @  @  @  @  @"
6050 PRINT"▒▒    A  A  A  A  A"
6080 PRINT"▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒  B "
6090 PRINT" CDE▒"
6100 RETURN
6999 REM**MOVE ALIENS**
7000 PRINTAM$;
7010 X=INT(4*RND(1)+1):FORI=1TOX:PRINT"▒▒";
     :NEXT
7020 IFDI=1ANDA%(X)<5THEN7100
7030 IFDI=-1ANDA%(X)>1THEN7200
7040 DI=-DI:GOTO7300
7100 PRINTCHR$(148):A%(X)=A%(X)+1:GOTO7300
7200 PRINT"▒";CHR$(20):A%(X)=A%(X)-1
7300 REM
7500 RETURN
7999 REM **MOVE PLAYER**
8000 GETYM$:IFYM$<>"P"ANDYM$<>"@"ANDYM$<>"*"
     THENRETURN
8005 IFYM$="@"ANDGH<>5THEN8080
8010 IFYM$="P"ANDCO>0THENCO=CO-1:GOTO8040
8020 IFYM$="*"ANDCO<16THENCO=CO+1:GOTO8040
8030 RETURN
8040 FU=FU-.075
8050 PRINTLM$;TAB(CO)"  B  ▒▒▒▒▒▒ CDE ";
8070 RETURN
8080 BP=8120+CO+2:BC=BP+30720:FU=FU-.35
8090 GOSUB7000
8095 IFPEEK(BP)=0ORPEEK(BP)=1THEN8200
8100 POKEBP,6:POKEBC,0
8120 GH=5:GOSUB9000:GOSUB8000:GH=0:POKEBP,32
8130 BP=BP-44:BC=BC-44:IFBP<7790THENRETURN
8140 GOTO8090
8200 POKEBP,8:POKEBC,10:POKE36877,190:FORI
     =15TO0STEP-.2
8205 POKE36878,I+144:NEXT:POKE36877,0
```

```
8210 POKE36878,8:POKEBP,32:NA=NA-1:SC=SC+10
8220 PRINT"▒▒▒▒SCORE:"SC:RETURN
9000 IFFU<=0THEN10000
9010 PRINTSP$;LEFT$(FU$,FU+4)
9020 RETURN
9999 REM **FAIL**
10000 PRINTSP$:FORI=1TO1000:NEXT:PRINT"▒▒";
10005 POKE36879,170:POKE36878,15:POKE36869,
      240
10010 PRINT"▒▒▒      OUT OF FUEL     "
10015 FORI=1TO1000:NEXT
10020 POKE36874,215:FORI=1TO500:NEXT
10030 POKE36874,207:FORI=1TO375:NEXT
10040 POKE36874,219:FORI=1TO125:NEXT
10050 POKE36874,215:FORI=1TO500:NEXT
10060 POKE36874,207:FORI=1TO500:NEXT
10070 POKE36874,0
10072 PRINT"▒▒YOU SCORED"SC"POINTS"
10080 PRINT"▒ANOTHER GO ? (Y/N)":POKE198,0
10090 GETA$:IFA$<>"Y"ANDA$<>"N"THEN10090
10100 IFA$="Y"THENRUN
10110 POKE650,32:POKE36879,27:PRINT"▒▒";:END
10160 GOTO10080
10999 REM **SUCCESS**
11000 FORI=1TO1000:NEXT
11005 POKE36878,15
11010 PRINT"▒":FORI=128TO254:POKE36876,I
11015 POKE36879,I:NEXT:POKE36879,93:
      POKE36869,240
11020 PRINT" ▒▒     WELL DONE      "
11030 PRINT"▒▒▒  YOU SHOT ALL THE    "
11040 PRINT"▒     ALIENS IN ▒"
11050 PRINTMID$(TI$,3,2)" MINUTES "
      :PRINTRIGHT$(TI$,2)" SECONDS▒"
11060 GOTO10080
12000 POKE36879,107
12010 PRINT"▒▒"TAB(5)"▒▒────────── "
12020 PRINTTAB(5)"▒▒▒▒ ▒ ▒ ▒▒ ▒▒ "
12030 PRINTTAB(5)"▒▒▒ ▒ ▒ ▒ ▒▒▒ "
12040 PRINTTAB(5)"▒▒▒ ▒ ▒ ▒ ▒▒▒ "
12050 PRINTTAB(5)"▒▒ ▒ ▒ ▒ ▒▒▒ "
12060 PRINTTAB(5)"▒▒▒ ▒ ▒ ▒ ▒▒▒▒▒ "
12070 PRINTTAB(5)"▒▒▒ ▒ ▒ ▒ ▒▒ ▒▒▒ "
12080 PRINTTAB(5)"▒▒────────▒▒▒"
12090 PRINTTAB(4)"▒SPACE INVASION▒"
12100 PRINT"▒   ▒BY ANDREW CASSON"
12110 FORI=1TO2500:NEXT
13000 RETURN

READY.
```



Memory Snapshot



The prolific Mr. R. W. Moore of Sutton Coldfield offers this short but really useful program. It displays the contents of memory, both in hexadecimal and in decimal form, and it's particularly handy when machine code is being POKEd into top of memory since it enables you to examine the memory area to check that your code has been entered correctly.

When you put in the required memory start (line 30) the subroutine at 1200 with subroutine 2000 converts the decimal value of the location into hexadecimal equivalents and puts it into string form in line 1330. Line 1031 examines the contents of memory location by location for 20 locations and subroutine 2000 converts the observed decimal content into hexadecimal for the higher byte; line 1033 does the same for the low byte. In line 1034 the low and high bytes are combined into a composite string and line 1040 prints the decimal and hexadecimal equivalents of the address and the contents on to the screen.

Line 1050 enables the next or proceeding blocks to be viewed, an entirely new start to be made or the program to be ended.

```
10 REM MEM
20 REM RWM 1982
1000 PRINT"■■SHOWS MEMORY CONTENTS IN BLOCKS
     OF 20"
1010 P=19:Q=0:PRINT"■MEMORY START (NORMALLY
     4096)?":INPUTV
1020 PRINT"■":FORT=V+QTOV+P
1030 GOSUB1200
1031 W=PEEK(T):N=INT(W/16):GOSUB2000
1032 G$=E$
1033 N=W-N*16:GOSUB2000
1034 H$=E$:J$=G$+H$
1040 PRINT "■";T;"■";PEEK(T);TAB(11)J$;"■";
     TAB(16);F$;"■":NEXT
1050 PRINT" ■";"R";"■";"ESTART";"■+■";",";"
     ■-■";"OR";"■E■";"ND"
1052 GETO$:IFO$=""THEN1052
1054 IFO$="R"THEN1010
1056 IFO$="+"THENP=P+20:Q=Q+20:PRINT"■":GOT
     O1020
1057 IFO$="E"THEN1100
1058 IFO$="-"THENP=P-20:Q=Q-20:PRINT"■":GOT
     O1020
1059 GOTO1050
1100 PRINT"■■END":END
1200 P1=256*16:P2=256:P3=16
1210 N1=INT(T/P1):N=N1
1220 GOSUB2000
1230 A$=E$
1240 N2=INT((T-N1*P1)/P2):N=N2
1250 GOSUB2000
1260 B$=E$
1270 N3=INT((T-N1*P1-N2*P2)/P3):N=N3
1280 GOSUB2000
1290 C$=E$
1295 N9=N1*P1+N2*P2+N3*P3
1300 N4=T-N9:N=N4
1310 GOSUB2000
1320 D$=E$
1330 F$=A$+B$+C$+D$
2000 IFN<10THENE$=CHR$(N+48):RETURN
2010 IFN=10THENE$="A":RETURN
2020 IFN=11THENE$="B":RETURN
2030 IFN=12THENE$="C":RETURN
2040 IFN=13THENE$="D":RETURN
2050 IFN=14THENE$="E":RETURN
2060 IFN=15THENE$="F":RETURN
2070 RETURN
```

AMERICAN SOFTWARE FOR YOUR VIC

Great VIC Software

New Programs for your VIC Computer On Cassette!

ALL PRICES INCLUDE VAT @ 15% AND POSTAGE/PACKING

COLOR & SOUND

ANY 3 FOR £28.50

- Adventure
- Head On
- Target
- Hang In
- Chase
- Cattle Roundup
- Artillery Shoot
- Micro Maze
- Snake Out
- Trapper

£10.25 EACH

- Lunar Lander
- Wrap
- Breakout
- Black Jack
- Pong
- Slots
- Alien Raiders
- Bombs Away
- Home Finance
- Math Drill
- Hangman
- Yahtze
- Othello
- Target Moon
- Shooting Gallery
- Capture

£10.25 EACH

EGG ATTACK JOYSTICK

This seemingly innocent electronic game of catch turns gradually into a hurried race of joystick control. The player must maneuver a nuclear reactor in order to catch radioactive eggs as they are dropped from advancing "monsters" If the player misses three eggs, the Earth is no longer habitable and their turn of play is over. **12.95**

TUNNEL PATROL JOYSTICK

The player is required in this arcade game to maneuvre an android through a maze in order to harvest "Energy Crystals" and score points. As an obstacle the program randomly throws the android into the fourth dimension where it must battle the dreaded "Batwing" in order to survive. **12.95**

SEND 75p FOR FULL CATALOGUE (Refundable against purchase)
DEALER ENQUIRIES WELCOME

ANGLO AMERICAN SOFTWARE
138a Stratford Road, Sparkhill
Birmingham B11 1AG 021-771 2995/2736

ALL PRICES INCLUDE VAT & POSTAGE
24 Hour answering service on 021-771-2995 for Access orders.

Please supply .....

.....

I enclose a cheque/PO for £..... **made payable to ANGLO AMERICAN or debit my**

Access card number .....

Signature .....

NAME .....

ADDRESS .....

.....

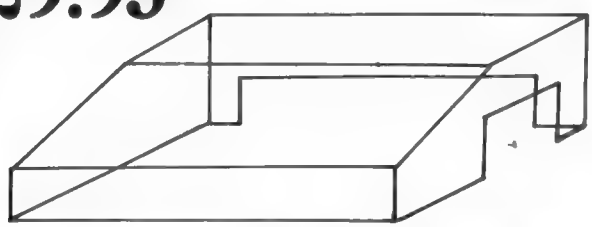
.....

BBC·VIC-20·ATARI·TANDY

WE'VE GOT YOU COVERED

£9.95

V–20



Is your computer gathering dust?
Do you want to keep it looking brand new?
Is it childproof or coffee stained?
We supply a moulded — gold tinted — Perspex — Custom Made — hard cover to protect and enhance the appearance of your computer, for only £9.95. A small price to pay for permanent protection!

PROTECT YOUR INVESTMENT

Please send me the items marked: I enclose £ .....

NAME: _____

ADDRESS: _____

_____

Detach and make cheques and Postal Orders payable to **SOFTCELL** Ltd.
(Dealer enquires welcome)

SOFTCELL LIMITED
87 HAMBLETON ROAD
HALESOWEN, WEST
MIDLANDS. B63 1JT
021-550 5063

- ☐ ATARI 400
- ☐ ATARI 800
- ☐ VIC 20
- ☐ BBC
- ☐ TANDY (Level II CPU)

Tick appropriate box.
£9.95 + £1.00 p&p.

222 PCW

Portfollo

From James Clunie of Whitley Bay comes a program that displays the performance of specific Stock Exchange shares graphically and in colour.

The computer reads the name and starting price for the share, then the prices for each of the nine weeks; and it works out the percentage increase or decrease in price for each week. It prints the results in the form of a bar chart — a blue bar for a gain, a red bar for a loss, and a message in black corresponding to 'no change'.

```
10 REM PORTFOLIO BY JAMES CLUNIE 20/1/82
20 READ N$:IF N$="XXX"THEN 999
30 PRINT "■":PRINT"*INVESTMENT  GRAPHICS
*":PRINT
40 PRINT"    **************    ":PRINT
50 READ S
60 PRINT"REVIEW OF LAST 9 WEEKS":PRINT
70 PRINT"STARTING PRICE:";S;"P":PRINT
80 FOR T=1TO9
90 READ P
100 LET D=P-S:IF D=0 THEN 300
110 LET D=D/S
120 LET D=D*100:IF D<0 THEN 200
130 FOR G=1TOD
140 PRINT;"■";
150 NEXT G
160 PRINT ,N$;" ";P;"P"
170 NEXT T
180 FOR W=1TO9000:NEXT W
190 GOTO 10
200 FOR G=1TO ABS(D)
210 PRINT;"■";
220 NEXT G
230 PRINT ,N$;" ";P;"P"
240 NEXT T
250 FOR W=1TO9000:NEXT W
260 GOTO 10
300 PRINT "■NO CHANGE";
310 PRINT ,N$;" ";P;"P"
320 NEXT T
330 FOR W=1TO9000:NEXT W
340 GOTO 10
400 DATA "BP"
410 DATA 310
420 DATA 322,332,326,318,312,316,298,300
,304
500 DATA "GEC"
510 DATA 752
520 DATA 760,781,782,808,814,829,800,806
,830
600 DATA "P&O"
610 DATA 130
620 DATA 127,123,125,119,124,128,130,138
,144
700 DATA "XXX"
999 END
```

Russian Roulette

Here are three more fom the highly active Mr R. Martin of Grimsby. **Russian Roulette** is the conventional game, but at least there's a wry twist or two to it: when the gun doesn't fire, the victim (VICtim, geddit?) smiles. And when it does, you get a few bars of the Death March.

```
5 XX=6
10 POKE36879,76
20 PRINT"■":PRINT"■■■   RUSSIAN ROULETTE
■"      :PRINT:PRINT
21 PRINT"■■ A GUN WITH 6 CHAMBERS":PRINT:P
RINT:PRINT"■■   IS POINTED AT YOU  ":PRINT:
PRINT
22 PRINT"■■THERE IS ONLY 1 BULLET":PRINT:PR
INT:  PRINT"■■ THERE WILL BE 6 SHOTS■
":PRINT:PRINT
23 PRINT"■■    HIT F TO START    "
24 GETA$:IFA$=""THEN24:IFA$="F"THEN40
25 IFA$<>"F"THENPRINT"■■■       COWARD!!!
■":IFA$<>"F"THENPOKE36879,8
26 IFA$<>"F"THENEND
39 Z=6
40 POKE36879,76:PRINT"■■■■■ ■■  ■ "
50 PRINT"■■■■ ■■■..■■  SHOTS LEFT";Z
55 Z=Z-1
60 PRINT"■■■■■■■  ■■■■"
70 PRINT"■■■■ ■■^■■ ■"
80 PRINT"■■■■ ■■  ■■ ■"
90 PRINT"■■■■■■■ ■■■":PRINT"■■■■■■■■"
95 PRINT"■■■■ ■■■■ "
100 PRINT"■■■■■     ■■■■■    ■■■■■     ■■
110 PRINT"■■■■■     ■■■■■    ■■■■■     "
120 PRINT"■■■■■    "
130 PRINT"■■■■■|  ■■■■■ |  ■■■■■  |  ■■■■■|"
140 PRINT"■■■■■■|■"
150 PRINT"■■■■■■|■■■■|■■■■|"
170 PRINT"■■■■■ |  "
180 PRINT"■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■.
190 PRINT"■■■■■■■■■■■■■■■■■■■■___■"
200 PRINT"■■■■■■■■■■■■■■■■■■■■■\/■ ■■"
210 PRINT"■■■■■■■■■■■■■■■■■■■■■■■■■■ ■"
215 FORT=1TO4000:NEXT
220 A=INT(RND(1)*6)
230 IFA=1 THENGOSUB1000
240 IFA<>1 THENGOSUB2000
250 GOTO40
1000 POKE36876,0:POKE36877,200:FORL=15TO0ST
EP-1
1005 RESTORE
1010 POKE36878,L
1020 NEXTL
1040 FORX=7715+198  TO7708+198 STEP-1
1050 POKEX,69
1060 POKEX+30720,0
1065 POKEX+1,32
1070 FORT=1TO50:NEXTT
1120 NEXTX
1122 POKEX+1,160:POKEX+1+30720,2:POKEX+24+3
0720,2:POKEX+24,46
1125 POKEX+25+44,46:POKEX+44+25+307
20,2
1130 POKEX+25+44+44,46:POKEX+25+44+44+3072
0,2
1140 POKEX+44+25+44+44,46:POKEX+25+44+44+4
4+30720,2
1150 POKEX+44+44+25+44+44,46:POKEX+25+44+4
4+44+44+30720,2
1160 POKEX+44+44+44+25+44+44,81:POKEX+44+2
5+44+44+44+44+30720,2
1170 POKE36877,0
1180 READN,S
1190 POKE36878,15
1200 IFN<0THEN1260
1210 POKE36875,N
1220 FORD=1TO40*S
1230 NEXTD
1240 POKE36875,0
1250 GOTO1180
1260 POKE36878,0
1300 PRINT"■":POKE36879,25
1310 PRINT"■■         YOU'RE  ":PRINT:PRINT
:PRINT:PRINT:PRINT:PRINT:PRINT
1320 PRINT"■■ ■■■■■   ■■■■■ ■"
1330 PRINT"■■ ■■■■■■■   ■■■■■■■ ■■■■"
1340 PRINT"■■ ■■■■■■■ ■■■■ ■■■ ■■ ■■■■■"
1350 PRINT"■■ ■■■■■■■ ■■■■ ■■■ ■■ ■■■■■"
1360 PRINT"■■ ■■■■■■    ■■    ■■ ■■■■■■"
1370 PRINT"■■ ■■■■■■    ■■    ■■ ■■■■■"
1380 PRINT"■■ ■■■■■■ ■■■■ ■■■ ■■ ■■■■"
1390 PRINT"■■ ■■■■■■ ■■■■ ■■■ ■■ ■■■"
1400 PRINT"■■ ■■■■■■    ■■ ■■■ ■■ ■■■"
1410 PRINT"■■ ■■■■■■    ■■ ■■■ ■■ ■■"
1490 PRINT"■"
1500 GOTO5010
2000 POKE36878,10:POKE36875,200
:FORT=1TO10:NEXTT:POKE36878,0
2010 PRINT"■■■■■■■■■■■■■■■v":PRINT:PRINT:PRINT
2020 IF Z=0THENGOSUB5000
2030 PRINT"■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■    "
2040 PRINT"■■■■■■■■■■■■ YOU SURVIVED ■":PRIN
T"■■■■■■■■■■■"
2050 PRINT"■■■■■■■■■■ ...THIS TIME ■"
2060 PRINT"■■■■■■■■■■■              ■"
3000 FORT=1TO5000:NEXT
3010 RETURN
5000 FORT=1TO2000:NEXT:             PRINT"
■■■■■■■■■■■■■■YOU SURVIVED THE LOT■"
5010 PRINT"■■HIT F TO PLAY AGAIN"
5020 GETA$:IFA$=""THEN5020:IFA$="F"THEN39
5030 GOTO25
10000 DATA175,6,175,6,175,3,175,6,189,6,18
4,3,184,6,175,3,175,6,169,3,175,6,-1,- 1
```

Photo Acoustics Ltd.

★ VIC 20 SPECIALS ★

(ALL PRICES INC. VAT)

VIC 20 + CASSETTE DECK + 16K RAM	£299.00
VIC 20 + CASSETTE DECK + 8K RAM	£273.00
VIC 20 + CASSETTE DECK + AFRON EXPANSION UNIT INC. LID	£325.00
VIC 20 + CASSETTE DECK + INTRO. TO BASIC PT. 1 + 10 BLANK CASSETTES + FITTED 13AMP PLUG	£243.00
VIC 20 + CASSETTE DECK + ROM CARTRIDGE	£250.00
VIC 20 + CASSETTE DECK + 1 GAMES TAPE	£232.00

HARDWARE

AFRON EXPANSION + LID	£99.00
16K RAM PACK	£72.00
8K RAM PACK	£44.00
3K RAM PACK	£39.00
HI-RES CARTRIDGE	£34.00
PROGRAMMERS AID CART.	£34.00

FULL RANGE OF HI-TECH SOFTWARE AND VIC CART'S FROM £6.99-£19.00

58, HIGH STREET, NEWPORT PAGNELL
BUCKS. TEL: 0908 610625

CREDIT CHARGE HP + MAIL ORDER

255A ST. ALBANS RD. WATFORD
HERTS. TEL: 0923 32006

VIC 20 IN SCOTLAND

PERIPHERALS-UTILITIES-SOFTWARE

Minidisc Drive	396.00	Line Printer	229.00
Arfon Expansion	99.00	Machine Code ROM	34.95
3K RAM	29.95	Prog. Aid ROM	34.95
8K RAM	44.95	Hi. Res. ROM	34.95
16K RAM	74.95	Vickit II	44.95
Stack Expansion Board with 3K RAM and Vickit I			73.20
RS 232C	55.95	Paddles (Pair)	13.50
Joystick	7.50	Light Pen	28.75
Multiplexer	37.95	Dustcover	2.25

SOFTWARE ROMS

Avenger-Superslot-Super Lander-Road Race all	19.95

Tape Software

Vicalc-Advanced calculator with 10 Memories	8.99
Telephone Directory-Create, ammend optional print	8.95
Skymath and Space Division-Maths for younger children	ea. 6.99
Simple Simon-Colour and Sound Simon game	6.99
A-maz-ing-Munchmen/pacman	6.99
Alien Blitz-Space invaders	7.99

Books

VIC Revealed	10.00
Programmers Reference Guide	14.95
Getting Acquainted (Revised)	5.95

NEW SOFTWARE ARRIVING ALL THE TIME. SEND FOR FULL LIST

BEM BUSINESS & ELECTRONIC MACHINES

7 Castle St, Edinburgh EH2 3BB Tel 031-226 5454

To BEM 7, Castle Street, Edinburgh EH2 3BB. Tel. 031-226 5454.

ORDER FORM from NAME _____

ADDRESS _____

QUANTITY	DESCRIPTION	PRICE
	Post FREE OVER £100.00 Add Post £2.50 (50p per item software)	
	TOTAL	

I enclose my *cheque/charge my Access/ Visa No. _____

Signature _____

CARD HOLDERS 'PHONE YOUR ORDER -24 HOUR SERVICE. *Delete as applicable

Mon-Sat

Graphics FOR YOUR PET/VIC

'HI PAD'
Digitising area: 11" x 11"
Size: 1" H x 17" W x 14" D
Data Rate: Up to 300 co-ordinate pairs/second
Modes: Point, switch stream, stream
Attachments: Crosswire cursor, stylus
Scaling: Provided in software

'MICROPLOT'
Plotting area: 180mm(X) x 250mm(Y) A4
Plotting speeds: 400 mm/sec Pen up; 70 mm/sec Pen down, variable
Resolution: ±0.05 mm
Interface: IEEE, Address 7 (can be changed)
Character set: Alphanumeric, £, π, ↑
Character size: 5mm – 30mm height
Axis drawing: Command for size and number of segments
Rotation: 0°, 90°, 180°, 270°
Basic Command set: 15

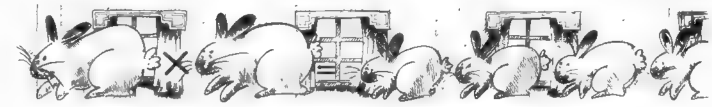
'DIGIPLOT'
Plotting area: 360mm(X) x 260mm(Y), A3
Plotting speed: 50 mm/sec
Resolution: 0.1mm
Interface: User Port (can be IEEE)
Character set: 96 ASCII characters, inc. Upper/Lower Case
Scaling: 16 sizes for printing characters
Axis drawing: Command for size and number of segments
Rotation: Rotate command for printing characters
Basic Command Set: 14
Can be 1 or 6 pen

See us at the SHOW

BUSINESS ELECTRONICS
'The Microcomputer Specialists'
OFFICIALLY APPROVED BY COMMODORE

ROWNHAMS HOUSE, ROWNHAMS, SOUTHAMPTON SO1 8AH
TEL SOUTHAMPTON (0703) 738248

Multiply



From Philip Mitchell of Milnrow, Greater Manchester comes a computer-aided learning program designed for young children (his six-year-old son in particular) who are beginning to learn the principles of multiplication.

The instructions are brief; but then laborious detail is useless to children of this age and help from an adult on the first run or two is much better.

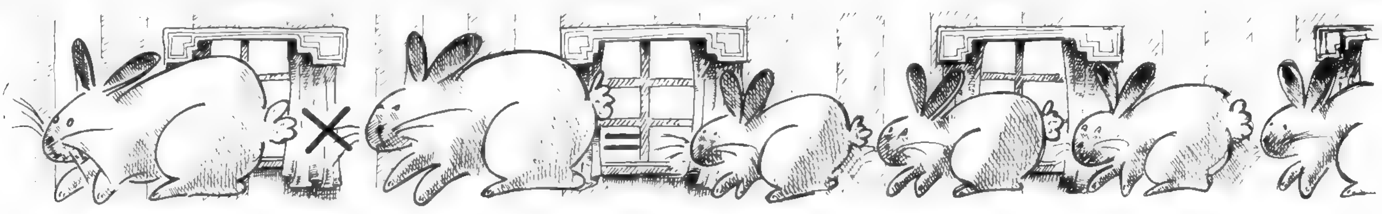
The first display asks for the factors required in the program. This is meant for the parent or teacher. The second factor can be set to a maximum so that products can be limited. The first factor can be constant, thus concentrating on one set of multiples; or it can be random — up to the maximum set for the second factor.

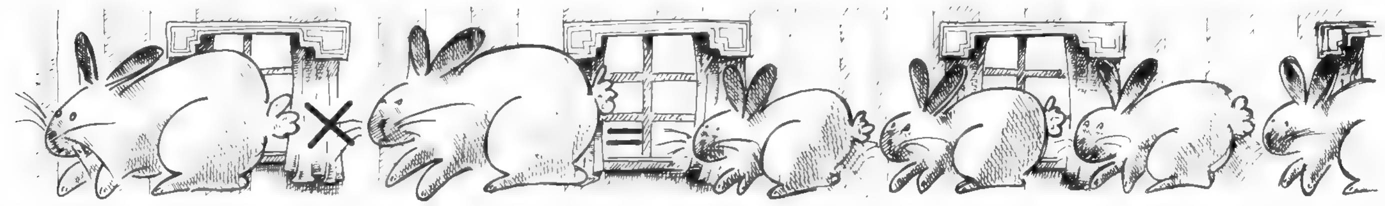
The program gives five examples showing notation, meaning and a pictorial representation of the product. If incorrect figures are entered, the child is returned to the beginning and it is all explained again. A test at the end is of the same five examples which are 'marked' as it goes along. Full marks are rewarded with a jingle.

The program originally gave the choice of "having another go" but Philip's son apparently insisted on being able to go back and change the range of numbers himself. Lines 541-544 can simply be changed to YES/NO choice again.

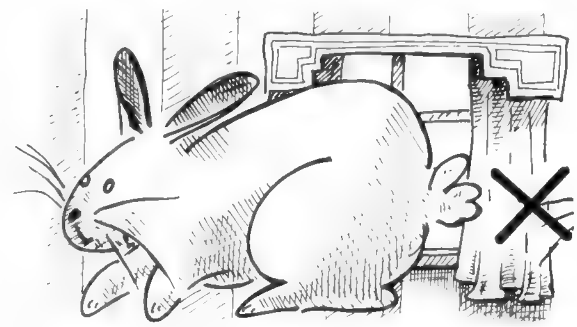
Simple but effective — though the program trips up if you try inputting too large a number. Maybe there should be a regulator in there to control the maximum size?

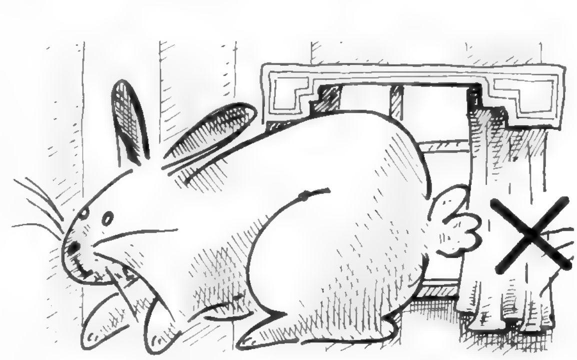
```
100 REM"*** MULTIPLY ***"
110 PRINT"■"
120 POKE36879,221
130 SV=36878:SO=36876:CC=0
140 PRINT"MAX. 2ND FACTOR":INPUTF1
150 PRINT"■■1ST FACTOR (IF RANDOM TYPE 0)"
160 INPUTF2:PRINT"
170 IFCC=1THENGOTO200
180 PRINT"■■■■■■■■■WHAT IS YOUR NAME"
190 INPUT"■■";NA$
200 FORI=1TO5
210 B(I)=INT(RND(1)*F1)+1
220 A(I)=F2:IFF2=0THENA(I)=INT(RND(2)*F1)+1
230 PRINT"■RIGHT "NA$
240 PRINT"■READ THE NOTES AND DO"
250 PRINT"WHAT IT TELLS YOU"
260 PRINT"■■■■■■                    ■"
270 PRINT" ■"A(I)"■MULTIPLIED■ ■BY■"B(I)"
280 POKE7879,160:POKE38599,7:POKE7897,160:POK
E38617,7
290 PRINT" ■                    ■  ";
300 PRINT" ■       ■IS■ ■WRITTEN■     "
310 PRINT" ■                    "
320 PRINT" ■       ■"A(I)"X"B(I)"■       ■"
330 PRINT" ■                    ■"
340 PRINT"■■■■PRESS<SPACE>TO GO ON"
350 GETGT$:IFGT$=""THEN350
360 PRINT"■"
370 PRINT"■■■■■■■■■■■■              ■"
380 PRINT" ■       ■"A(I)"X"B(I)"■       ■"
390 PRINT" ■                    ■"
400 PRINT" ■          ■MEANS■        ■"
410 PRINT" ■                    ■"
420 PRINT" ■    ■"A(I)",■ ■"B(I)"TIMES■
 ■"
430 PRINT" ■                    ■"
440 PRINT"■■■■PRESS<SPACE>TO GO ON"
450 GETGT$:IFGT$=""THEN450
460 PRINT"■"
470 PRINT"■■■■                  ■"
480 PRINT" ■       ■"A(I)"X"B(I)"■       ■"
490 PRINT" ■                    ■"
500 PRINT" ■  ■CAN■ ■BE■ ■SHOWN■ ■AS
■  ■"
510 PRINT" ■
520 POKESO,128+10*I
530 PRINT:PRINT
540 POKESV,10
550 FORP=1TOA(I)
560 POKESO,128+10*I
570 PRINT"■■■";
580 FORQ=1TOB(I)
590 PRINT"■ ●";
600 NEXTQ
610 PRINT
620 POKESO,0
630 NEXTP
640 POKESV,0:POKESO,0
650 PRINT"■■■";
660 FORR=1TOB(I)
670 PRINT"■"A(I)"■";
680 NEXTR
690 PRINT:PRINT
700 PRINT"NOW...HOW MANY SPOTS  IS THAT
ALTOGETHER"
710 INPUTS
720 IFS<>A(I)*B(I)THEN GOSUB1060:GOTO460
730 GOSUB1140
740 PRINT"■■■■■■■■■■■NOW THEN"
750 PRINT"■■CAN YOU REMEMBER IT   WITHOUT THE
DRAWING?"
760 PRINT"■■■■■■■■■"A(I)"X"B(I)"=";:INPUTS1
770 IFS1<>A(I)*B(I)THENGOSUB1110:GOTO230
780 IFS1=A(I)*B(I)THENGOSUB1090
790 NEXTI
800 PRINT"■■■NOW "NA$:PRINT"WE'LL HAVE A LITTLE
 TEST"
810 PRINT"■■JUST YOU TYPE THE     ANSWERS!"
820 SC=0
830 FORI=1TO5:PRINT"■"A(I)"X"B(I)"=";:INPUTS2
840 POKESV,15
850 IFS2<>A(I)*B(I)THENGOTO900
860 IFS2=A(I)*B(I)THENPRINT"
 ■√■■":SC=SC+1
870 POKESO,235
880 FORDE=1TO500:NEXT
890 POKESO,0:GOTO940
```





```
900 IFS2<>A(I)*B(I)THENPRINT"
    "
910 POKESO+1,135
920 FORDE=1TO500:NEXT
930 POKESO+1,0
940 NEXTI
950 POKESV,0
960 PRINT"YOU SCORED"SC"OUT OF 5"
970 IFSC=5THENGOSUB1280
980 PRINT"DO YOU WANT ANOTHER GO"
990 INPUTY$:IFLEFT$(Y$,1)="N"THEN GOTO1040
1000 IFLEFT$(Y$,1)="Y"THENPRINT"SAME AGAI
N";:INPUTYY$
1010 IFLEFT$(YY$,1)="Y"THENGOTO200
1020 IFLEFT$(Y$,1)="Y"ANDLEFT$(YY$,1)="N"T
HENCC=1:GOTO140
1030 GETY$:IFY$="Y"THENGOTO200
1040 PRINT"O.K. 'BYE "NA$:END
1050 GOTO1030
1060 PRINT"ARE YOU SURE?"
1070 PRINT"JUST COUNT THEM AGAIN":GOSUB1
250
1080 FORDE=1TO1000:NEXT:RETURN
1090 PRINT"CORRECT--WELL DONE"
1100 GOSUB1190:FORDE=1TO2000:NEXT:RETURN
1110 PRINT"NO, SORRY, THAT'S NOT RIGHT..
."
1120 PRINT"LET'S HAVE ANOTHER  LOOK AT
THIS ONE...."
1130 GOSUB1250:FORDE=1TO5000:NEXT:RETURN
1140 PRINT"":POKE36879,43
1150 PRINT"CORRECT"
1160 PRINT""A(I)"X"B(I)"="S:GOSUB11
90
1170 FORDE=1TO2000:NEXT
1180 POKE36879,221:PRINT"":RETURN
1190 POKESV,15
1200 FORL=1TO50:POKESO,INT(RND(1)*128)+128
1210 FORM=1TO10
1220 NEXTM:NEXTL
1230 POKESO,0:POKESV,0
1240 RETURN
1250 POKESV,15:POKESO+1,135
1260 FORDE=1TO1000:NEXT
1270 POKESV,0:POKESO+1,0:RETURN
1280 POKESV,10
1290 READP:IFP=-1THEN1350
1300 READD
1310 POKESO,P
1320 FORDE=1TOD:NEXT
1330 POKESO,0
1340 GOTO1290
1350 POKESV,0
1360 DATA228,140,223,280,223,140,223,140,
219,140,223,140,225,520,223,280,223,140
1370 DATA219,280,219,140,219,140,215,140,
219,140,223,520,215,280,219,140,223,280
1380 DATA223,140,223,140,219,140,223,140,
225,280,228,140,231,280,231,140,228,280
1390 DATA225,140,223,280,219,140,215,560,
-1
1400 RESTORE:RETURN
```





Split the Cards

In **Split the Cards** you bet on the value of the middle card being between than of the other two. You must bet, but you aren't allowed to be more than you have in the bank — which is broken when it reaches 1000.

```
5 B=63:BA=50:GOSUB8900
10 PRINT"■"
20 B=63:S=0:AA=INT(RND
(1)*13)+1
30 IFAA=1THENA=50
31 IFAA=2THENA=51
32 IFAA=3THENA=52
33 IFAA=4THENA=53
34 IFAA=5THENA=54
40 IFAA=6THENA=55
41 IFAA=7THENA=56
42 IFAA=8THENA=57
43 IFAA=9THENA=20
50 IFAA=10THENA=10
51 IFAA=11THENA=17
52 IFAA=12THENA=11
53 IFAA=13THENA=1
60 CC=INT(RND(1)*13)+1
65 IFCC=1THENC=50
70 IFCC=2THENC=51
75 IFCC=3THENC=52
80 IFCC=4THENC=53
85 IFCC=5THENC=54
90 IFCC=6THENC=55
95 IFCC=7THENC=56
100 IFCC=8THENC=57
105 IFCC=9THENC=20
110 IFCC=10THENC=10
115 IFCC=11THENC=17
120 IFCC=12THENC=11
125 IFCC=13THENC=1
130 IFCC=AATHEN60
140 GOSUB5000
150 INPUT"■
■PLACE YOUR STAKE■";S
155 IFS>BATHEN150
157 IFS<1THEN150
160 BA=BA-S
170 PRINT"■
■":PRINT"■";BA
180 FORT=1TO1000:NEXTT
200 BB=INT(RND(1)*13)+1
210 IFBB=1THENB=50
220 IFBB=2THENB=51
230 IFBB=3THENB=52
240 IFBB=4THENB=53
250 IFBB=5THENB=54
260 IFBB=6THENB=55
270 IFBB=7THENB=56
280 IFBB=8THENB=57
290 IFBB=9THENB=20
300 IFBB=10THENB=10
310 IFBB=11THENB=17
320 IFBB=12THENB=11
330 IFBB=13THENB=1
340 GOSUB5000
350 IFBB>=CCANDBB>=AATHENPRINT"■LOST■"
360 IFBB<=CCANDBB<=AATHENPRINT"■LOST■"
370 IFAA<BBANDBB<CCTHENGOSUB8000
380 IFAA>BBANDBB>CCTHENGOSUB8000
390 PRINT"■      "
400 PRINT"■";BA
410 FORT=1TO3000:NEXT
412 IFBA>999THEN2000
415 IFBA<=0THEN1000
416 PRINT"■          "
417 PRINT"■DO YOU WANT    ■
■TO PLAY AGAIN?"
430 GETXX$:IFXX$=""THEN430
440 IFXX$="Y"THEN10
450 POKE36879,27:PRINT"■COWARD!!":END
1000 POKE36879,27:PRINT"■BROKE!!■":END
2000 POKE36879,27:PRINT"■YOU BROKE■
■THE BANK!■"
2010 PRINT"■PLAY AGAIN?"
2020 GETCC$:IFCC$=""THEN2020
2030 IFCC$="Y"THENRUN
2040 GOTO450
5000 POKE36879,106
5005 PRINT"■                    "
5010 POKE8164,109:POKE8165,64:POKE8166,64:POKE8167,64:POKE81
68,64:POKE8169,125
5020 POKE8142,93:POKE8120,93:POKE8098,112:POKE8147,93:POKE81
25,93:POKE8103,110
5030 POKE8102,64:POKE8101,64:POKE8100,64:POKE8099,64:POKE812
1,2:POKE8122,1:POKE8 123,14
5040 POKE8124,11
6010 POKE7711,79:POKE7712,99:POKE7713,99:POKE7714,99:POKE771
5,80
6015 POKE7708,80:POKE7707,99:POKE7706,99:POKE7705,99:POKE770
4,79
6020 POKE7718,79:POKE7719,99:POKE7720,99:POKE7721,99:POKE772
2,80
6030 POKE7726,101: POKE7748,101:POKE7770,101:POKE7792,101:PO
KE7814,101:POKE7836, 101
6040 POKE7858,101:POKE7880,101:POKE7902,101:POKE7924,76
6050 POKE7925,100:POKE7926,100:POKE7927,100:POKE7928,122:POK
E7931,76:POKE7932,10 0
)60 POKE7933,100:POKE7934,100:POKE7935,122:POKE7938,76:POKE
7939,100:POKE7940,10 0
6070 POKE7941,100:POKE7942,122:POKE7920,103:POKE7898,103:POK
E7876,103:POKE7854,1 03
6080 POKE7832,103:POKE7810,103:POKE7788,103:POKE7766,103:POK
E7744,103
```

NEW PROGRAMS FOR THE UNEXPANDED VIC 20

PRC1 WAR, SMASHOUT, BLACKJACK, LOGIC, PICKUPGAME, ALARM CLOCK Our 6 program best selling introductory cassette

PRC2 FLY SNATCHER
Superb colour and sounds as the weasel, who has an unfair advantage, chases you. Your only escape is to eat the flies and dive into a bush!

PRC3 ANT RAIDERS
The ants very gracefully shuffle around the screen while you try to shoot them. Easy? Just try it!

PRC4 SUB ATTACK/MISSILE COMMAND
2 great, noisy, colourful games. Sub Attack – a big fun torpedo firing range! Missile Command – fire to destroy the mutants while dodging their deadly touch!

EACH CASSETTE IS ONLY £7.50 INCLUSIVE

Please send ☐ PRC1 ☐ PRC2 ☐ PRC3 ☐ PRC4

I enclose cheque/PO for £_____

NAME _____

ADDRESS _____

P R SOFTWARE
28 THE FAIRWAY, SOUTH RUISLIP, MIDDLESEX HA4 0RY

VIC COLOUR COMPUTERS £197

CATCH UP WITH TODAY'S WORLD OF MICROCOMPUTERS ON THIS LOW COST POWERFUL MODEL FROM COMMODORE
IT HAS SOUND, COLOUR, EXPANDABILITY
A COMPREHENSIVE COMPUTER LANGUAGE.
A GREAT PROGRAMMING BOOK . . . IN FACT EVERYTHING YOU NEED!

VIC20 COMPUTER **£197.**
PROGRAM RECORDER **£44.95.**
TOGETHER **£240.**

WE HAVE THE RANGE OF ACCESSORIES AND SOFTWARE.

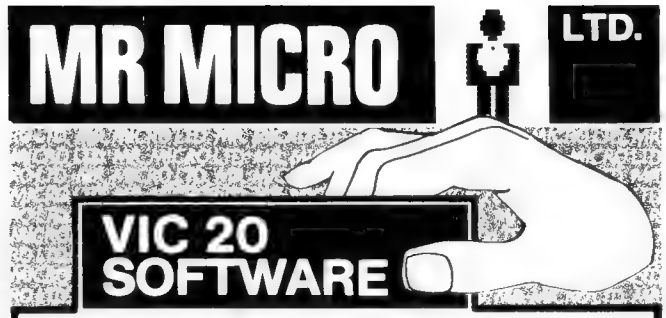
BUY YOUR VIC FROM THE EXPERTS
THE PEOPLE WHO CARE!
THE PEOPLE WHO KNOW!

WE GIVE 12 MONTHS FULL GUARANTEE

PCP

PERSONAL COMPUTER PALACE
4-6 CASTLE STREET, READING, BERKSHIRE.
Telephone: (0734) 589249

MR MICRO LTD.



VIC 20 SOFTWARE

VIGIL THE NEW 60 COMMAND GAMES LANGUAGE
Includes **9** games: Breakout, Anti Aircraft, Space War, Concentration, Maze, U.F.O., Space Battle, Kaleidescope and Fortune Teller.
50 Page manual included.
NOW ONLY £25
Requires 3K expansion.

GRAPHVICS ADD 18 NEW COMMANDS
152 x 160, Hi-Res & Multicolour on same screen. Save/load Pictures. Shows you exactly what your VIC is really capable of. Complete with sample programmes and manual.
£18.95
3K or 8K expansion required.

MYSTERIOUS ISLAND
JULES VERNE GRAPHIC VIC VENTURE
HIGH RESOLUTION HIGH ENJOYMENT.
Pack includes: Mastertape, Practice Progs, Extra Data Tape and Play Guide.
£16
Requires 16K expansion.

GOLD RUSH The amazingly successful PRIZE VIC VENTURE
For the Prospector who finds the Golden Word. Tape and Guide.
PRIZE NOW OVER £275
£16

GREAT BALLOON RACE
THE FUN GRAPHIC RACE GAME
Kitty now over £125.
Current High Score 644 by C. Church of Essex.
Every entrant has 3 chances to win.
£16

ASTRO COMMAND Fire your missiles using Joystick Guidance before your cities are destroyed. **£6.95**

VIC TREK boldly go where no VIC has gone before! **£6.95**

DAMSEL Up to 6 players follow the path to rescue the Damsel, overcoming Ogres & Monsters on the way. **£6.95**

VIC MEN The BugByte version of PACMAN. **£6.95**

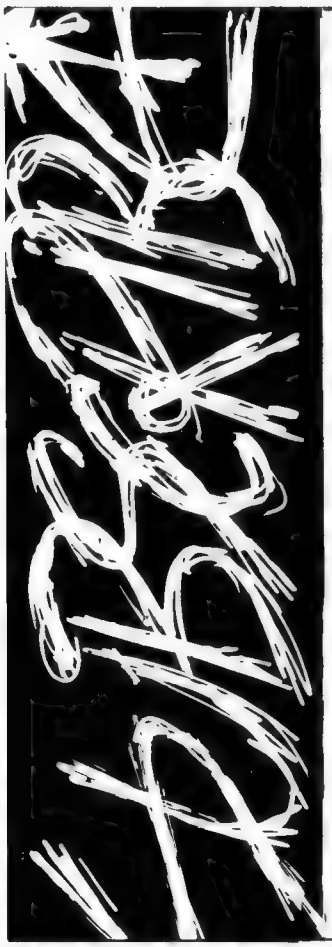
VIC VALUE No1 ONLY **£7.95**
Four great games one one tape! Helicopter Lander, Vic Hunter, Dragon and Alien Pilot.

SPECIAL OFFER
For the months of June and July.
Buy any of the above programs and you can have POTSHOT and RAINBOW PASSAGE one one tape
for only £2.95

ALL PRICES INCLUDE VAT & P. & P.
Order by Telephone with Barclaycard or by cheque/P.O. made payable to Mr. Micro Ltd.
NEW CATALOGUE: OVER 40 PROGS. SEND LARGE SAE
MR. MICRO LTD
P.O. Box 24, Swinton, Manchester, M27 3AL
Tel: 061-728 2282 FOR VISA ORDERS

```
6090 POKE7740,101:POKE7762,101:POKE7784,101:POKE7806,101:POKE7828,101:POKE7850,1 01
6100 POKE7872,101:POKE7894,101:POKE7916,101:POKE7913,103
6110 POKE7891,103:POKE7869,103:POKE7847,103:POKE7825,103:POKE7803,103:POKE7781,1 03
6120 POKE7759,103:POKE7737,103:POKE7733,101:POKE7755,101:POKE7777,101:POKE7799,1 01
6130 POKE7821,101:POKE7843,101:POKE7865,101:POKE7887,101:POKE7909,101
6140 POKE7906,103:POKE7884,103:POKE7862,103:POKE7840,103:POKE7818,103:POKE7796,1 03
6150 POKE7774,103:POKE7752,103:POKE7730,103
6200 POKE38447,2:POKE7727,83:POKE38454,2:POKE7734,83:POKE38461,2:POKE7741,83
6210 POKE38449,0:POKE7729,88:POKE38456,0:POKE7736,88:POKE38463,0:POKE7743,88
6220 POKE7903,65:POKE38623,0:POKE7910,65:POKE38630,0:POKE7917,65:POKE38637,0
6230 POKE7905,90:POKE38625,2:POKE7912,90:POKE38632,2:POKE7919,90:POKE38639,2
6500 POKE7816,A:POKE7823,B:POKE7830,C
7000 PRINT"[graphics]"BA    :PRINT"[graphics]"
7001 RETURN
8000 BA=BA+2*S:PRINT"[graphics]WIN";2*S
8020 RETURN
8900 POKE36879,249
9000 PRINT"[graphics]                                SPLIT THE CARDS
 "

9010 PRINT"[graphics]                                "
9020 PRINT"[graphics]                                "
9030 PRINT"[graphics] TWO CARDS ARE DEALT"
9040 PRINT"[graphics]YOU HAVE A BANK OF 50 "
9050 PRINT"[graphics] YOU MUST BET ON THE"
9060 PRINT"[graphics]   THIRD CARD BEING"
9070 PRINT"[graphics]BETWEEN THE OTHER TWO"
9080 PRINT"[graphics]    YOU BREAK THE      "
9090 PRINT"[graphics]     BANK AT 1000      "
9100 PRINT"[graphics] HIT ANY KEY TO PLAY  "
9200 GETDD$:IFDD$=""THEN9200
9300 RETURN
```



L & J COMPUTERS FOR VIC-20

WE DON'T BOAST HOW BIG WE ARE.....
BUT WE DO BRAG ABOUT OUR SUPER SERVICE!
ALL ORDERS RECEIVED ARE DESPATCHED THE SAME DAY
WHETHER BY CHEQUE, ACCESS OR VISA.
(CHEQUES ARE SUBJECT TO VERIFICATION)

WE HAVE A SUPER RANGE OF SOFTWARE AVAILABLE EX-STOCK
+ BLANK C12 & CASE @ £4.50 FOR 10 INC. VAT. (ADD £1 POST)

SEND US A LARGISH STAMPED ADDRESSED ENVELOPE FOR
FREE SOFTWARE & HARDWARE LISTS

192 HONEYPOT LANE, STANMORE MIDDX.
01-204 7525 HA7 1EE

Taylor Wilson
Consumer Electronics Division

The West Midlands VIC-20 COMPUTER SPECIALISTS

Full range of Commodore peripherals, software and accessories
Station Road, Dorridge, Solihull, West Midlands, B93 8HQ.
Tel: 05645 6192. Saturday Line: 05645 70229.

ADVANCED BUSINESS COMPUTERS LTD.

If you live in the West Midlands
then visit our Vic 20 showrooms.
Large range of Vic games and
accessories always in stock!
Expert advice.

Sutton Computer Centre,
26 High Street,
Sutton Coldfield,
West Midlands.
Tel 021 354 2684

Micro Business Centre Ltd,
Linthouse Lane,
Wednesfield,
Wolverhampton.
0902 725687

CHROMASONIC electronics

COME AND MEET VIC
at

YOUR NORTH LONDON VIC CENTRE

48, JUNCTION ROAD, ARCHWAY, LONDON N19 5RD
Telephone: 01-263 9493/01-263 9495
Telex no. 22568

BUSINESS ELECTRONICS

PET. APPLE. VIC-20. PRESTEL

Micro-Magic Ground Floor
Debenhams Southampton

ROWNHAMS HOUSE, ROWNHAMS,
SOUTHAMPTON SO1 8AH
Telephone: SOUTHAMPTON (0703) 738248

TYPEWRITER CENTRES

for all your VIC requirements
Branches throughout the MIDLANDS and BOURNEMOUTH
PET specialists - COMMODORE appointed dealers

BIRMINGHAM		
City Centre:	52 Bromsgrove Street	021 622 5385
Edgbaston:	92 Dudley Road (opp. hospital)	021 455 9111
Erdington:	52 High Street (opp. library)	021 382 0185
Kings Heath:	46 High Street	021 444 7439
Sutton Coldfield	Knights House, Gracechurch Centre	021 355 6789
LEICESTER:	34-36 Rutland Street	0533 538745
WALSALL:	7 Leicester Street	92 27589
WOLVERHAMPTON:	Worcester Street	90 27627
COVENTRY:	128 Far Gosford Street	0203 28799
BOURNMOUTH:	75 Seamoor Road, Westbourne	0202 766997

ANIROG Computers

LARGE SELECTION OF SOFTWARE
GAMES — BUSINESS — EDUCATIONAL

VIC Centre
CO-OP CRAWLEY

AR

MAIL ORDER — 26, Balcombe Gardens
Horley, Surrey
Horley 2007/6083

HOLDENE LTD.

See
VIC 20
at

BRAY HOUSE
LEICESTER PLACE, LEEDS 2 LS2 9BH

Terminal Software

Games Cassettes for *Unexpanded VIC-20*
TVIC1 Line Up 4 & Panic Driver **£4.99**
TVIC2 Maze Muncher & Reversi **£4.99**

Two Games per cassette — Machine Code Action
More soon !!!
28 Church Lane, Prestwich, Manchester M25 5AJ

The Vic Computer Centre

South East Television Services
7 Spippal Street,
Dartford, Kent.
(0322) 27668

The instant character editor - Part two

The programmable character editor is a tool with which you can set up your very own characters, symbols and Blue Meanies on the Vic screen. Part One of our exposition appeared in the last issue along with a listing of the program.

We showed you how to use the editor to create user-defined characters: what we didn't tell you then was how you can incorporate those special symbols in your own programs. Read on.

It's easy enough to SAVE your newly created character set on a blank tape. Before you can use it, though, you'll have to get rid of the editor itself — which is still sitting there in memory, and it's a pretty large program.

After you've created your characters, it makes sense to remove the editor program and keep only the character set you've created, so you can use the memory space to write a longer Basic program around your new characters.

So once you've stored the characters you could turn the Vic off and on, LOAD the new character set, and RUN it. Alternatively, try this:

```
POKE 51,0: POKE 52,28: POKE 55,0: POKE 56,28: NEW
```

and hit RETURN. Now try a program which uses those special characters. Type Q for QUIT to get out of the character display and key in

```
POKE 51,0: POKE 52,28: POKE 55,0: POKE 56,28: NEW
```

Write your program with the new characters. For example, if you've redefined the bracket keys (character numbers 27 and 29) you would write a program in Basic using bracket keys where you want your newly created symbols and shapes to appear.

You also have to include on the first line (or wherever the special' characters appear) a special POKE which tells the Vic to use your special characters:

```
10 POKE 36869,255
20 FOR X=1T022: PRINT "[]"*NEXT
```

When the program finishes you'll still be in the special character mode: so if you type brackets now, the special characters will appear. To get back to normal, type POKE 36869,240.

So the two important commands are POKE 36869,255 to get you into your special characters: and POKE 36869,240 to return to normal. Obviously both of these will work if you type them in directly or use them in a program line.

And if you want to be unsubtle, another way to get back to normal mode is to hold down RUN/STOP and hit RESTORE.

Saving 'special character' programs

There are three pretty safe methods of saving Basic programs that use your special characters. For a start, here's the cleanest method and probably the easiest to use for the beginner. It does take some time: but when it's completed you need no special actions to use the program.

Essentially, what you're going to do is list out the eight-bit codes for each special character and copy them manually into your program in a form the Vic can understand. This is actually how most of the professionally-produced cartridge-based programs are created and tested before they are put into machine code.

So LOAD the editor (oh come on, you did remember to SAVE it, didn't you?) and create your first 'special' to replace one of the standard characters — it could be any character, but for this example we'll substitute the '@' sign. Now type SAVE, hit RETURN, and when the Vic asks you SAVE OR LIST? answer 'L'. If you've got a printer, you can list the DATA statements out on it; otherwise you'll have to copy them down manually.

The DATA statements (five numbers in each line) will then be displayed on your screen, starting with character number 0 and continuing through the whole set. The first number is always the number of the character: the other four numbers are the DATA entries you need to create the character in a program.

Now turn your Vic off and on (or type SYS64802 and RETURN). Key in this program:

```
2 IF PEEK (52)=28 THEN 7
3 POKE 51,0: POKE 52,28: POKE 55,0: POKE 56,28: clr: CB=7168
4 READ A: IF A=1-1 THEN 7
5 FOR N=0 TO 7: READ B: POKE: CB+A*N, B: NEXT
6 GOTO 4
7 POKE 36869,255
10 PRINTCHR$(144) "@@": FOR X=1 TO 100: NEXT:
PRINTCHAR$(156) "A": GOTO 10
100 DATA 0,255,189,219,255,255,195,189,255
```

This will SAVE and LOAD like any normal Basic program. The first seven lines let you use the special character(s): the DATA statements define the first (zero) character which is the '@' sign.

More DATA lines can be used to define more characters. For example, here's a second DATA line to redefine character number 1 (the letter A) in just the same way:

```
200 DATA 1,255,189,219,255,255,195,189,255,-1
```

Line 10 is the customised part and can be any Basic program. Simply use the '@' sign or the CHR$ number to use it.

There are several uses for this technique. One is to redefine the entire letter/number character set as foreign language characters — and some budding entrepreneur among the readers of **Vic Computing** might profitably market foreign language character sets for the Vic-20 using this editor to create them, by writing a program to redefine the entire character set with DATA statements.

The other standard approach is to design a game using your very own graphics. In addition to the one-character graphics we've shown, you can also create multiple-character shapes by setting up several characters to be printed together as a single large-size image. In other words, you can also create multi-character images on a larger scale.

Another SAVE technique

Method number two adds characters to an existing Basic program easily enough — but it does require that the original character set and the Basic program are on separate tapes. Load your program from tape, and add this as its first line:

```
1 POKE 45,AAA: POKE 46,BBB: POKE 1,0: POKE 52,28: POKE 55,0: POKE 56,28: clr
```

Now find the end of the program by typing PRINT PEEK (45), PEEK (46) in direct mode. Replace the 'AAA' in line 1 with the results of PEEK (45); and replace the 'BBB' with the results of PEEK (46). Make sure you hit RETURN after you change the line, to enter it.

Take the tape on which you SAVEd your special characters and put it in the player, type LOAD " ",1,1 and RETURN.

To SAVE the entire Basic program with the character set you can type this (in direct mode,

DISP
Used to set up the display area. The Vic will place a solid cursor on each of the display squares in turn, and ask you what character you wish to display in that area. Answer with a number from 0 to 63, and hit RETURN. That character will be displayed in that space until you change it with a subsequent DISPlay command. As an aid, the information line will show the character already under the cursor when DISPlay is typed: so hitting the RETURN key will leave that character still displayed. This allows you to see what characters are presently defined in the display area.

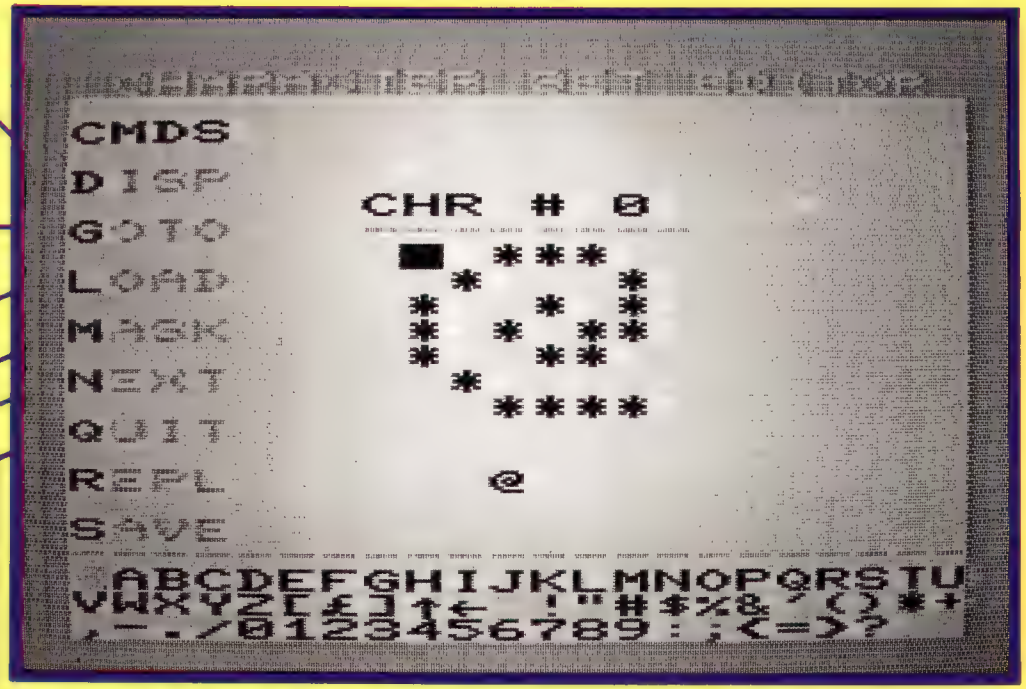
GOTO
Allows you to go to any character (from 0 to 63). When you type 'G', the Vic will ask you what character number you want.

LOAD
Loads a previously saved character set for further modification — the character information would have been stored on tape or disk by the SAVE command. It is loaded beginning in the 512 byte block starting at location 7168. To load from disk, specify a file name beginning with '&'.

MASK
Specifies a 'mask' for an individual character. When you type 'M' the Vic will ask you what character set you want and what character you would like. Type the set identifier (0 = upper case/graphics, 1 = upper/lower case) and the character number (0 to 127). The Vic will change your current character into the new character. You can then modify it as you choose.

NEXT
This switches you to the next character in sequence.

QUIT
Exits the program.



Editing commands

Key	Function
LEFT	Move cursor one space to the left
RIGHT	Move cursor one space to the right
UP	Move cursor one space up
DOWN	Move cursor one space down
*	Place a dot at the current cursor location
SPACE	Clear a dot at the current cursor location
RVS	Reverse the current character
HOME	Home the cursor
CLR	Erase the current character

LIST
If the LIST option is selected, the Vic will ask if you want to list to the printer. If you answer Y (and have a printer connected) the DATA statements needed to POKE the character set into memory will be printed on the printer. Otherwise the DATA statements will be displayed on the screen, in groups of 10.

REPL
Similar to MASK, except that instead of MASKing from the characters in the character ROM in the Vic the characters you have already created are used. The Vic will ask you what character number you want: answer with a number from 0 to 63. The character you specify will be copied into your current character, replacing what is already there. You are then free to modify the copy, without affecting the original.

SAVE
Store the currently defined 64-character set on tape or disk as a 512 byte program file, with a header that specifies that it will always be loaded beginning at location 7168. The length of the file name must be less than five characters.

Dear Vic

Send us comments, queries and complaints: we'll answer everything we can and print anything that isn't boring or illegal. Address yourself to The Editor, Vic Computing, 39-41 North Road, London N7 9DP

I would like to first of all congratulate you on a great April addition (sic). I am writing to ask about expanding my Vic. If I buy an Arfon expansion unit will I be able to plug in a 20K RAM cartridge? I want to use as few of the sockets as possible for RAM. The reason for this is to leave as many spaces as possible for ROM, like the Toolkit and Hi-Res cartridges.

Colin Murphy, Liverpool 28

As far as we know, anything designed to plug into the standard expansion slot will also fit the Arfon board (or for that matter the Stack expander unit). So you should be ok.

I was somewhat disturbed to see, in the April issue of **Vic Computing**, a LIFE program apparently submitted to you by Mr R W Moore of Sutton Coldfield.

The program bears an uncanny resemblance to a LIFE program written by me, and included in my book **Symphony for a Melancholy Computer.** Mr Moore appears to have taken my program, changed the display so that it uses POKE rather than PRINT, then submitted it to you for publication. He has not changed the variable names, or even cleaned up my somewhat erratic line numbering.

I am aware that it is very difficult for editors to know when material which is submitted for publication has been plagiarised, and do not believe you infringed my copyright knowingly.

However, as it has been done, I would like (a) an acknowledgement that my program has been printed by you; (b) some compensation for the infringement; and (c) perhaps you could issue a general warning to your contributors to be careful of the sources of their programs.

Tim Hartnell, London W8

R W Moore's LIFE does indeed look and read like Tim Hartnell's. If that is the case, it's certainly reprehensible. Technically, taking someone else's program and passing it off as your own is infringement of copyright or 'theft of intellectual property': to us it sounds like stealing.

There's a couple of grey areas between creativity and misusing another's work. For one thing, there's nothing new under the sun: statistically it's not improbable that two solutions to the same problem (like two programs for the same game) will take the same approach, use the same kind of structure, employ similar patches of code.

But then there's the more pressing problem — the case of converting or extending a program originally written by someone else. If you change a line or two, does that make the program yours? If no, just how many changes do you have to make before it stops being someone else's program and starts being your own work?

Lawyers get rich arguing about this kind of thing. Ultimately though, it all comes down to a matter of personal ethics. If you believe honestly that you have adapted a program such that it is sufficiently changed to make it recognisably yours, that's probably the best we can hope for.

As well as the programs in books like Tim Hartnell's, several of the computer magazines run programs received from readers — as happens with our own Victuals section in Vic Computing. We expect you to key them in for your own use. We also expect that you'll have fun modifying and amending them. We don't expect that you will subsequently send a submission to Victuals that bears a strong resemblance to the original — and if you send in something that is based on a program we have already printed, we expect you to acknowledge that fact.

We have had unkind things to say about another of Tim Hartnell's Vic-related publications in the past, but that's neither here nor there. He is entitled to the same kind of respect, protection and honest dealing as the next person. And, for the record, we understand that (a) Symphony for a Melancholy Computer is a considerable improvement errors-wise; and (b) Getting acquainted with your Vic is undergoing substantial revisions to clear out the bugs. Incidentally, we have Symphony under review.

COMPUTER USERS AIDS has been running the UK101 User Group now for some two years and is currently branching out into Video Genie as well as the ACORN ATOM and ZX81 fields. Obviously we see the VIC as an excellent first time buyer machine as well as a 'real computer' in its own right.

We found that a User Group provides a source of unbiased information and help for the user that can really only be gained from experience with the machine. The personal approach with help and advice is also very important to the new Computer owner who may find him/herself confused with the number of available additions, programs etc.

By keeping up to date with the latest additions, price changes and the like we are able to offer the user a unique brand of help that will enable them to get the best from their machine. The new VIC User Group will be expected to receive members from all parts of the country, as well as other parts of the world, Sweden, Iceland etc. where there are a large number of interested VIC users.

Further details about the User Group are available if your readers require. Many thanks for your help.

Adrian Walters, Computer User Aids, Romford

Interested parties can reach Mr Walters at 14 Carlton Road, Romford, Essex RM2 5BD.

I have acquired a Vic 20, which is somewhat unfortunate as to me computers are monumentally boring machines.

However, having got it, I thought I might as well make use of it for minor business purposes such as address lists, price lists, stock records, and any other simple function which might become apparent.

A check on Commodore's current advertisement showed that it specifically states that a full range of software for business is available on disc, cassette and cartridge.

But, having purchased some 20 computer magazines, the only remotely businesslike facility I've been able to find is a cassette entitled "Telephone Directory".

Last month I even took the day off work to visit The Computer Fair at Earls Court (something of a shambles incidentally) specifically to enquire after the advertised Vic 20 'business' software.

In the event the Vic 20 seemed to be rather poorly promoted at this show. I fought my way to the counter of one Stand (at which I bought the first copy of your magazine I'd seen). But the chap there gave me only a pitying look when I enquired about business software and muttered something about learning to program the thing myself, or buy a Pet — two things I've no intention of doing (and should not have to, if Commodore's advertisement is truthful).

On the other hand, the ZX-80/81 people had quite a range of business programs on offer for their toy computer, some of which were demonstrated to me on request.

Naturally these programs weren't quite up to the standards of the true professional ones. Nevertheless, they could obviously fulfil a number of useful marginal functions for small firms. Indeed, the range of software of all sorts for the Sinclair machines far outstripped those of all other brands.

It's beginning to seem to me that I should perhaps exchange this Vic 20 for a ZX-81 (or, now, a ZX Spectrum) — or pocket the cash I get for it and use it for increasing the number of my annual flying hours. A most attractive thought!

Generally I'm feeling pretty frustrated and angry and if I'd actually had to pay for this Vic 20 myself I'd be considering reporting Commodore to the Advertising Standards Authority.

Can you help? If so please do. If not then at least say so!! Do **not** recommend me contacting my local Commodore Agent. There are reasons why I do not wish to do that.

J.O. Grohmann, Wallington, Surrey

P.S. As a 'quid pro quo' I enclose my annual subscription to **Vic Computing!!!**

He's right: there it is in the Commodore ad — "Full range of software for home, education, business and entertainment on disk, cassette and cartridge". Much of that seems a bit previous to us, actually — though perhaps the definition of what constitutes a 'full' range is a matter of personal opinion.

Now, we do know that Commodore is dead keen to get together some more software, and particularly programs for education and business. There are a number of people beavering away in both departments, and submissions from the likes of you and me are being solicited provided they're any good. Business-biased programs are starting to appear: at the least there's the modest but not dumb spreadsheet circulator VICALC, reviewed by us two issues back, and a couple of filing packages are on their way. But we don't know of any software for accounting or payroll, for instance; and as yet there is nothing at all available on Vic disks (apart from the tests and getting-started demo: not exactly a barrel of laughs).

So is the Vic a business computer? No, not yet: but it seems likely that some businesslike packages will be available for it by the end of the summer. But after all it was not really designed for that role in business: whatever the ads say, in our view it is best seen as a low-cost method of introducing yourself to computers, to programming, and perhaps to the possibilities of both.

robox

COUNT ON US

The Scottish Microcomputer and Calculator Centre

Units 14 & 15,
Anderston Centre,
Argyle Street,
Glasgow
Tel: 041 221 5401/2

Lichfield, Tamworth, Stafford, Stoke,
Burton on Trent

If you live in or near these towns
buy your VIC from the professionals

Contact either Robert Jones or Brian Homewood
at

PEACH DATA SERVICES LTD.

5 Horninglow Street,
Burton on Trent,
DE14 1NJ Telephone: 0283 44968

Yorkshire Electronics

COMMODORE APPOINTED
COMMERCIAL SYSTEMS AND VIC DEALER

FULL RANGE OF HARDWARE AND SOFTWARE
FULL CREDIT TERMS AVAILABLE

EYS

CAXTON HOUSE,
17 FOUNTAIN STREET, MORELEY,
WEST YORKSHIRE. Tel: 0532-522181

CURRIE & MAUGHAN
MICRO COMPUTERS
204 DURHAM ROAD,
GATESHEAD.

TEL.
GATESHEAD
774540

Sumlock
MANCHESTER.

Commodore appointed VIC 20 dealer.
Permanent showroom-shop display.
Hardware-Software-Accessories.
Books-Service-Maintenance.
Personal computers with Personal service.

198 DEANSGATE, MANCHESTER.
TELEPHONE 061 834 4233.

AUGHTON MICRO SYSTEMS

19 Princes Street
Merseyside

Telephone: Southport 33405

KINGSLEY COMPUTERS LTD
132 DESBOROUGH RD
HIGH WYCOMBE BUCKS. HP11 2PU
TEL (0494) 449749

AMPLICON

AMPLICON MICRO SYSTEMS LIMITED
SALES OFFICES AND SHOWROOMS AT:
143 DITCHLING ROAD BRIGHTON
EAST SUSSEX BN1 6JA U.K.
TEL: BRIGHTON (0273) 562163
RICHMOND ROAD BRIGHTON
EAST SUSSEX BN2 3RL U.K.
TEL: BRIGHTON (0273) 608331
TELEX: 877470 AMPCON G

NORTH EAST

KE

COMPUTER SERVICES LIMITED
Osborne House, 28 Osborne Road
Newcastle upon Tyne NE2 2AJ
Telephone (0632) 815157

ADVANCED COMPUTER EQUIPMENT (LEEDS) LTD
MICROCOMPUTER SYSTEMS
95 MEADOW LANE · LEEDS 11 · 0532-446960 · TELEX 335909

£175 **VIC 20 PLUS** £175

Our price includes VAT and Delivery Charges
13 Amp Plug fitted and fully tested
FULL AFTER SALE SERVICE & SUPPORT
Access and Barclaycard – Telephone your order

SALES – SERVICE – SATISFACTION

with no line number needed)

:POKE 45,0: POKE 46,30: SAVE "program name"

Your program and character set have now been SAVEd on the same tape — so try RUNNING your program. Note however that no further changes can be made to the program or character set; the program could be damaged.

SAVE technique no. 3

The third method allows you to create and SAVE your character set on one tape, write and SAVE a Basic program on a second tape, add a subroutine to the program that links the special characters to the program, and then LOAD both programs together on a third tape.

So produce your special characters and SAVe them. Remove the tape and insert a new one. Write a Basic program using the special characters and include the following LOAD subroutine:

```
1 IF PEEK (52)=28 THEN 4
2 POKE 51,0: POKE 52,28: POKE 55,0: POKE 56,28: M1=PEEK (45): M2=PEEK (46): LOAD " ",1,1
3 POKE 45,M1: POKE 46,M2: POKE 47,M1: POKE 48,M2: POKE 49,M1: POKE 50,M2: clr
4 POKE 36869,255
```

SAVE the Basic program on the fresh tape and remove it. DO NOT REWIND THE TAPE!

Now reset the Vic (off/on or SYDS64802), LOAD and RUN the editor, and LOAD the special character from tape one. Remove that tape and put tape two (the program tape) back into the cassette player. Key LOAD " ",1,1 and RETURN. Now type S to SAVE your programmable characters on the same tape as the program: they will be automatically LOADed immediately after the Basic program on the same tape with the program.

From this point on, if you load the program and type RUN it will run with the special characters in the program.

Multiple specials

You can use more than 64 programmable characters at a time by effectively creating three tapes of 64 characters each. POKE 36869,254 will switch to these characters, and here's a program that you can substitute for the method 3 example above to load the separate characters sets:

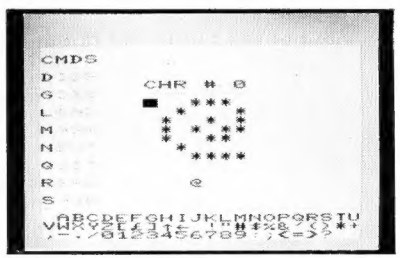
```
1 IF PEEK (52)=24 THEN 5
2 POKE 51,0: POKE 52,24: POKE 55,0: POKE 56,24: clr
3 LOAD " ",1,1: X=6144: GOSUB 5: X=6656: LOAD " ",1,1: GOSUB 5
4 LOAD " ",1,1: clr: GOTO 6
5 FOR M=0 TO 511: POKE M+X, PEEK (7168+M): NEXT: RETURN
6 POKE 36869,254
```

Using some standard characters

When using 64 or 192 programmable characters, the normal Vic characters are still available for your programs. When you POKE 36869,255 and the character set is changed to, for example, game graphics, you can still include upper-case characters in your programs by PRINTing the reversed upper case characters.

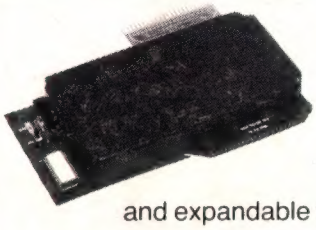
Using the Vic disk

The Vic 1540 disk naturally makes it easier (and quicker) to store and retrieve programs that feature user-defined characters. All the instructions given should work, except that filenames for SAVes and LOADs must be prefixed by an ampersand '&'.



VIC-20 ACCESSORIES FROM STACK

STACK STOREBOARD (memory expansion unit)



Power up your VIC-20 to a MASSIVE 32k COMPUTER!!

only **£49.00** (plus VAT) for 3k and expandable to 27k on the same board.

STACK LIGHTPEN

Allows you to use VIC-20 without keyboard entry by simple programming. Sensor in pen sees the TV screen! Ideal for education, games, menu selection etc.

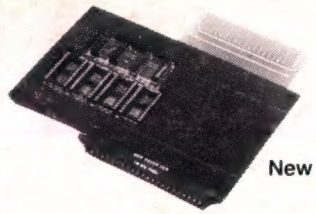
only **£25.00** (plus VAT)

STACK 8k RAMPACK

Use this upgrade pack to increase memory size on Stack Storeboard by 8k a time.

NEW LOWER PRICE!! only £29.00 (plus VAT)

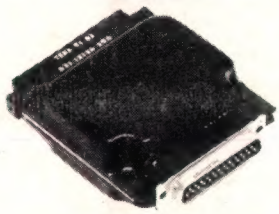
STACK ROM SWITCHBOARD



Use up to 4 ROMs at once! eg. games ROMs, VICKIT, VICKIT II etc.

New Lower Price!! **£29.00** (plus VAT)

STACK LOW COST RS232 INTERFACE



Allows you to use a serial printer with your VIC-20.

£22.99 (plus VAT)

STACK VICKIT SERIES

A series of ROMs which greatly simplifies programming and enhance the qualities of your VIC-20. Fits into Stack ROM SWITCHBOARD or Stack STOREBOARD.

VICKIT

Offers HELP to programmers....it also offers AUTO. DELETE. DUMP. FIND. OFF. RENUMBER. STEP. TRACE.

£25.00 (plus VAT)

Special Offer Price if Purchased with STOREBOARD only **£15.00** (plus VAT)

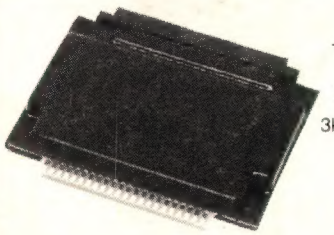
VICKIT II

A 4k ROM offering ALL THE FEATURES OF VICKIT plus.....TEXT. GRAPHICS. LINE. CLEAR. DRAW. PUT. FILL. SET. POINT

only **£29.00** (plus VAT)

Other exciting additions to the VICKIT series due soon!

STACK LOW COST 3k MEMORY



The lowest costing memory addition gives you 6½k of user memory on your VIC-20. Also allows you to use those quality games which demand 3k of Hi-Res Graphics! Socket at rear allows you to stack up further accessories.

only **£25.99** (plus VAT)

Contact your local Commodore VIC dealer for details.

Stack Computer Services Limited, 290-298 Derby Road, Bootle, Merseyside. 051-933 5511. Telex: 627026.